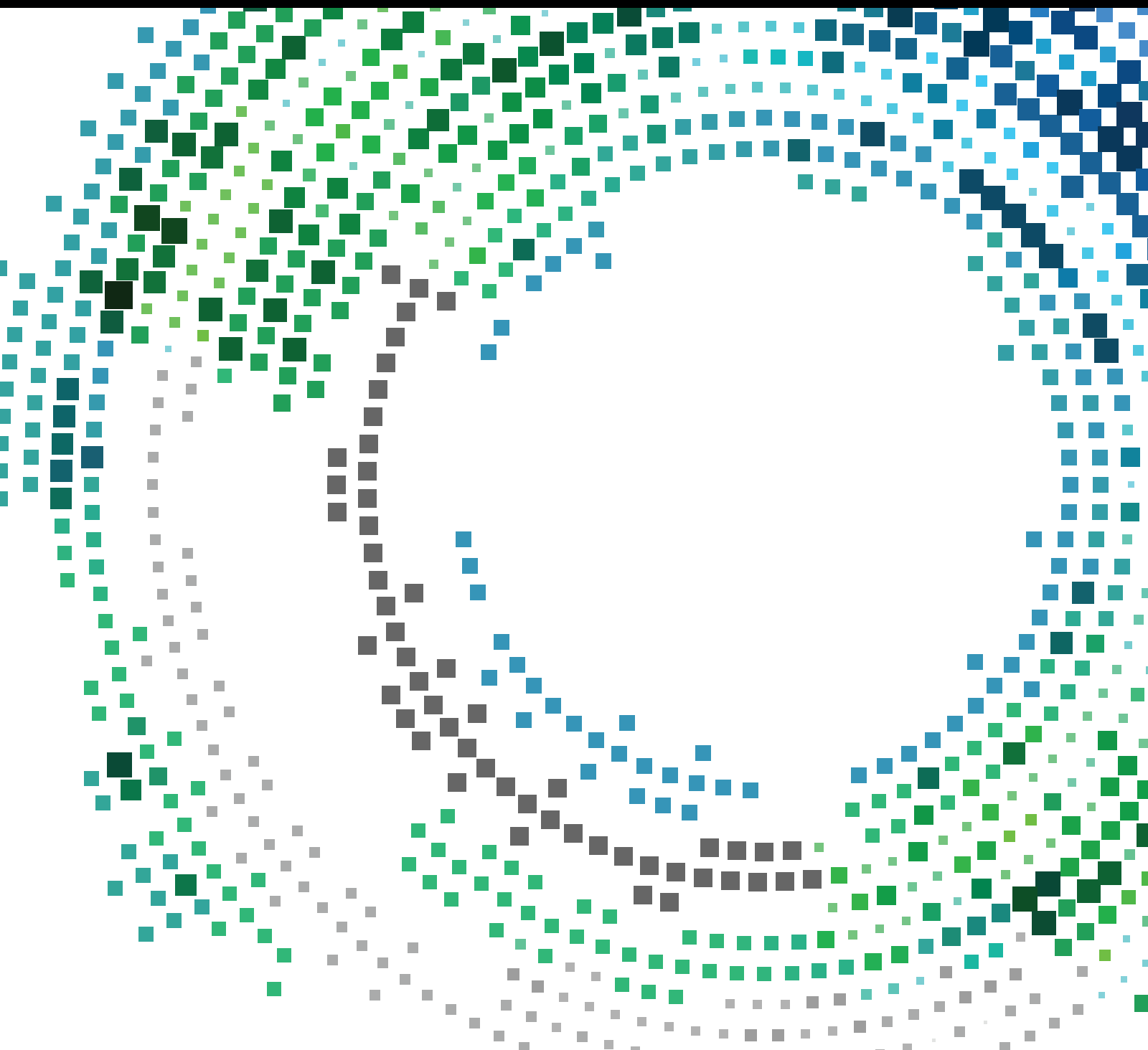
Mobile Information Systems

Internet of Everything

Lead Guest Editor: Laurence T. Yang
Guest Editors: Beniamino Di Martino and Qingchen Zhang



Internet of Everything

Mobile Information Systems

Internet of Everything

Special Issue Editor in Chief: Laurence T. Yang
Guest Editors: Beniamino Di Martino and Qingchen Zhang

Editorial Board

Contents

Hindawi
Mobile Information Systems
Volume 2017, Article ID 8035421, 3 pages
https://doi.org/10.1155/2017/8035421

Editorial
Internet of Everything

Laurence T. Yang,[1] Beniamino Di Martino,[2] and Qingchen Zhang[1]

[1]*St. Francis Xavier University, Antigonish, NS, Canada*
[2]*University of Campania "Luigi Vanvitelli", Caserta, Italy*

Correspondence should be addressed to Laurence T. Yang; ltyang@stfx.ca

Received 16 May 2017; Accepted 16 May 2017; Published 3 July 2017

Due to recent advancements in big data, connection technologies, and smart devices, our environment is transforming into an "Internet of Everything" (IoE). The Internet of Everything has become a catch-all phrase to describe adding connectivity and intelligence to just about every device in order to give them special functions. However, this can be quite reductive, as IoE provides links among not only things, but also data, people, and (business) processes. Evolution of current sensor and device networks, with strong interaction with people and social environments, will have a dramatic impact on everything from city planning, first responders, military, and health. Several Internet and connection-based paradigms fall under the IoE umbrella, such as Internet of Things (IoT), Internet of People (IoP), and Industrial Internet (II). While such areas cover many aspects of today's life, there is still the strong requirement to contextualize and integrate data and information coming from different networks and frameworks. Indeed, there is a need to provide a common ground for integrating information coming from heterogeneous sources. Such a shared ecosystem would allow for the interaction among data, sensor inputs, and heterogeneous systems.

Therefore, it is high time to call for contributions to further stimulate the continuing efforts to enable such an integrated framework; some fundamental issues are required to be addressed to provide the necessary bridge between different data representations and to solve terminology incongruence. This special issue attracted numerous submissions, out of which twenty papers have been accepted to be published. These accepted papers are expected to highlight some of the important aspects outlined above. It is the pleasure for the guest editorial team to introduce these papers as follows.

The paper entitled "Implementation and Optimization of GPU-Based Static State Security Analysis in Power Systems" by Y. Chen et al. contributes to static state security in power systems. A sensitivity analysis-based method with graphics processing unit (GPU) is proposed for power systems, which involves load flow analysis and sensitivity analysis. A multifrontal method for sparse LU factorization is explored on GPU in the load flow analysis while the varying matrix operations during sensitivity analysis on GPU are highly optimized.

The paper entitled "RAID-6Plus: A Comprised Methodology for Extending RAID-6 Codes" by M.-Z. Deng et al. focuses on RAID-6 code. In particular, a new RAID-6 code extending methodology with shorter reconstruction window is developed in this study, which provides a balanced tradeoff of flexible reliability and better system performance. Furthermore, an example extension code called RDP+ is presented based on RDP in terms of encoding and single failure reconstruction improvement. In order to validate and evaluate the presented extending methodology, a new metric called Q-metric is proposed.

In the paper entitled "Recommending Locations Based on Users' Periodic Behaviors" by B. Xu et al., the challenging topic of location recommendation based on user's life behavior is addressed. In view of multiple periodic behaviors existing in time series, an algorithm which can mine all periods in time series is proposed in this paper. Particularly,

locations using item-based collaborative filtering algorithm are recommended based on the periodic behaviors.

The paper entitled "Recovering Individual's Commute Routes Based on Mobile Phone Data" by X. Song et al., aims to propose a commute routes recovering model to recover individuals' commute routes based on passively generated mobile phone data, which applies two modules. The first is data preprocessing module, which extracts commute trajectories from raw dataset and formats the road network into a better modality. The second module combines two kinds of information together and generates the commute route with the highest possibility.

The paper entitled "A Process Mining Based Service Composition Approach for Mobile Information Systems" by C. Huang et al. investigates the connection between large scale of data and the associated business processes in the Internet of Everything (IoE) environment. Particularly, a process mining based service composition approach is proposed in this paper in order to improve the adaptiveness and efficiency of compositions. Firstly, a preprocessing is conducted to extract existing service execution information from server-side logs. Then process mining algorithms are applied to discover the overall event sequence with preprocessed data.

The paper entitled "Energy-Efficient Broadcasting Scheme for Smart Industrial Wireless Sensor Networks" by Z. Chen et al. proposes a novel energy-efficient broadcast scheme with adjustable broadcasting radius to improve the performance of network upgrade in smart industrial wireless sensor networks. In their scheme, the nonhotspots sensor nodes take full advantage of their residual energy caused in data collection period to improve the packet reception probability and reduce the broadcasting delay of code packet transmission by enlarging the broadcasting radius, that is, the transmitting power.

The paper entitled "Exploiting Wireless Received Signal Strength Indicators to Detect Evil-Twin Attacks in Smart Homes" by Z. Tang et al. focuses on Evil-Twin attack in smart home environments. In particular, this paper presents a novel Evil-Twin attack detection method based on the received signal strength indicator (RSSI). This approach considers the RSSI as a fingerprint of APs and uses the fingerprint of the genuine AP to identify fake ones. Furthermore, two schemes are presented to detect a fake AP in two different scenarios where the genuine AP can be located at either single or multiple locations in the property, by exploiting the multipath effect of the Wi-Fi signal.

In the paper entitled "An Extended Technology Acceptance Model for Mobile Social Gaming Service Popularity Analysis" by H. Chen et al., an empirical study on WeChat, China's most popular mobile social network, is presented. Particularly, a technology acceptance model (TAM) is applied to study the reasons beneath the popularity of games in mobile social networks. Furthermore, factors from social and mobile perspective are incorporated into the conventional TAM and their influence and relationships are studied.

The paper entitled "Power-Aware Resource Reconfiguration Using Genetic Algorithm in Cloud Computing" by L. Deng et al. aims to address the power-aware resource reconfiguration in cloud computing. In this paper, several algorithms for VM placement (multiobjective genetic algorithm (MOGA), power-aware multiobjective genetic algorithm (pMOGA), and enhanced power-aware multiobjective genetic algorithm (EpMOGA)) are presented to improve stability of VM placement pattern with less migration overhead. Furthermore, nondominated sorting genetic algorithm II (NSGAII) is used to select new generations during evolution process.

The paper entitled "Delay-Aware Program Codes Dissemination Scheme in Internet of Everything" by Y. Xuan et al. investigates a delay-aware program dissemination (DAPD) scheme to disseminate program codes with fast, reliable, and energy-efficient style. DAPD scheme improves the performance of bulk codes dissemination through the following two aspects. (1) Due to the fact that a high transmitting power can significantly improve the quality of wireless links, transmitting power of sensors with more residual energy is enhanced to improve link quality. (2) Due to the fact that performance of correlated dissemination tends to degrade in a highly dynamic environment, link correlation is autonomously updated in DAPD during codes dissemination to maintain improvements brought by correlated dissemination.

The paper entitled "Channel Selection Policy in Multi-SU and Multi-PU Cognitive Radio Networks with Energy Harvesting for Internet of Everything" by F. Hu et al. contributes to channel selection in a multi-SU and multi-PU cognitive radio network. In this paper, the authors adopt cooperative sensing method to avoid the packet collision between SUs and PUs and focus on how to collect the spectrum sensing data of SUs for cooperative sensing. Furthermore, they propose a competitive set based channel selection policy for multi-SU where all SUs competing for data transmission or energy harvesting in the same channel will form a competitive set.

The paper entitled "Making Image More Energy Efficient for OLED Smart Devices" by D. Li et al. focuses on energy consumption of OLED displaying in smart devices. In particular, this paper proposes an approach to improve image energy efficiency on OLED displays by perceiving image content. The key idea of this approach is to eliminate undesired details while preserving the region of interest of the image by leveraging the color and spatial information. First, we use edge detection algorithm to extract region of interest (ROI) of an image.

In the paper entitled "Speed-Density Model of Interrupted Traffic Flow Based on Coil Data" by C. Yu et al., a new method which can accurately describe the speed-density relation of interrupted traffic flow is proposed for speed fluctuation characteristics. The model of upper and lower bounds of critical values obtained by fitting the data of the coils on urban roads can accurately and intuitively describe the state of urban road traffic, and the physical meaning of each parameter plays an important role in the prediction and analysis of such traffic.

The paper entitled "A Novel Exercise Thermophysiology Comfort Prediction Model with Fuzzy Logic" by N. Jia et al. aims to address the prediction of exercise accidents in a regular exercise program. Particularly, a human thermophysiology regulatory model is designed to enhance the human thermophysiology simulation in the HCE system.

Some important thermal and physiological performances can be simulated. According to the simulation results, a human exercise thermophysiology comfort prediction method based on fuzzy inference system is proposed.

The paper entitled "Time-Aware IoE Service Recommendation on Sparse Data" by L. Qi et al. investigates the challenges of the recommendation technique on the service selection decision of target users. In view of the challenges, a time-aware service recommendation approach is proposed in this paper. Concretely, the time-aware user similarity is first calculated; afterwards, indirect friends of the target user are inferred by Social Balance Theory (e.g., "enemy's enemy is a friend" rule); finally, the services preferred by indirect friends of the target user are recommended to the target user.

The paper entitled "Congestion Control Mechanism for Intermittently Connected Wireless Network" by R. Wang et al. proposes a congestion control mechanism that is based on the network state dynamic perception. Specifically, through estimating the congestion risk when a node receives packets, ICWN can reduce the probability of becoming congested. Moreover, due to ICWN's network dynamics, the congestion risk threshold is determined by jointly taking into account the average packet size, average forwarding risk, and available buffer resources. Further, the service ability of a node in a distributed manner is evaluated by integrating the recommendation information from other intermediate nodes. Additionally, a node is selected as a relay node according to both the congestion risk and service ability.

The paper entitled "STLIS: A Scalable Two-Level Index Scheme for Big Data in IoT" by Y. Leng et al. focuses on index scheme for big data in IoT. In particular, this paper proposes a scalable two-level index scheme (STLIS) for RDF data. In the first level, a compressed path template tree (CPTT) index is proposed based on S-tree to retrieve the candidate sets of full path. In the second level, a hierarchical edge index (HEI) and a node-predicate (NP) index are created to accelerate the match.

In the paper entitled "An Indoor Ultrasonic Positioning System Based on TOA for Internet of Things" by J. Li et al., an ultrasonic indoor positioning system is presented, which can achieve centimeter-level precise positioning of objects moving indoors. The system is based on long-baseline positioning technology that uses code division multiplexing access mechanism. Particularly, the system uses wideband pseudorandom noise signal called Gold sequences for multiuser identification and slant range measurement.

The paper entitled "Phase Clustering Based Modulation Classification Algorithm for PSK Signal over Wireless Environment" by Q. An et al. proposes a new signal classifier for phase shift keying (PSK) signals. The periodicity of signal's phase is utilized as the assorted character, with which a fractional function is constituted for phase clustering. Particularly, an advanced estimator is proposed for estimating the frequency offset and balancing estimation accuracy and range under low signal-to-noise ratio (SNR) conditions.

The paper entitled "A High-Order CFS Algorithm for Clustering Big Data" by F. Bu et al. investigates the clustering scheme for big data in Internet of Things. In particular, this paper proposes a high-order CFS algorithm (HOCFS) to cluster heterogeneous data by combining the CFS clustering algorithm and the dropout deep learning model, whose functionality rests on three pillars: (i) an adaptive dropout deep learning model to learn features from each type of data, (ii) a feature tensor model to capture the correlations of heterogeneous data, and (iii) a tensor distance-based high-order CFS algorithm to cluster heterogeneous data.

Acknowledgments

The guest editorial team would like to thank all the authors for submitting their manuscripts to this special issue and all the invited reviewers for their time and constructive feedback.

Laurence T. Yang
Beniamino Di Martino
Qingchen Zhang

Hindawi
Mobile Information Systems
Volume 2017, Article ID 1897476, 10 pages
https://doi.org/10.1155/2017/1897476

Research Article

Implementation and Optimization of GPU-Based Static State Security Analysis in Power Systems

Yong Chen,[1,2] Hai Jin,[1] Han Jiang,[3] Dechao Xu,[2] Ran Zheng,[1] and Haocheng Liu[1]

[1]*Services Computing Technology and System Lab, Cluster and Grid Computing Lab, Big Data Technology and System Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China*
[2]*China Electric Power Research Institute, Beijing 100192, China*
[3]*Global Energy Interconnection Development and Cooperation Organization, Beijing 100031, China*

Correspondence should be addressed to Ran Zheng; zhraner@hust.edu.cn

Received 23 September 2016; Accepted 5 January 2017; Published 15 March 2017

Academic Editor: Beniamino Di Martino

Static state security analysis (SSSA) is one of the most important computations to check whether a power system is in normal and secure operating state. It is a challenge to satisfy real-time requirements with CPU-based concurrent methods due to the intensive computations. A sensitivity analysis-based method with *Graphics processing unit* (GPU) is proposed for power systems, which can reduce calculation time by 40% compared to the execution on a 4-core CPU. The proposed method involves load flow analysis and sensitivity analysis. In load flow analysis, a multifrontal method for sparse LU factorization is explored on GPU through dynamic frontal task scheduling between CPU and GPU. The varying matrix operations during sensitivity analysis on GPU are highly optimized in this study. The results of performance evaluations show that the proposed GPU-based SSSA with optimized matrix operations can achieve a significant reduction in computation time.

1. Introduction

An electric power system includes a network of connected electrical components which are used to supply, transfer, distribute, and use electric power. It is necessary to know the physical properties of the power system for safety. Different experiments can be conducted to study the properties of power system. Sometimes, however, it is impossible to perform such experiments on large-scale, complex power systems because the experiments are too expensive or difficult to take measurements on those power systems. Therefore, power system analysis [1] becomes another popular choice and plays more and more important role in planning, design, and operation of electrical power system. Power system analysis aims at building an equivalent model for a given power system, running load flow calculation, evaluating the effects when subjected to disturbances, and generating the ways to improve the stability performance of the power system. Power system analysis can be classified into *steady state analysis* and *transient stability analysis*. The former is further categorized into *Load flow analysis* and *Static state security analysis* (SSSA). Load flow analysis involves determining voltages and power flow through a stable system, where any transient disturbances are assumed to have settled down. SSSA involves power flow calculation when a chosen device is connected or disconnected. Transient stability is analyzed to estimate the system's stability under dynamic conditions during disturbances.

Load flow analysis is one of the most significant computations in power system planning and operations. In load flow analysis, we need to model the entire network with all generators, loads, transmission lines, transformers, and reactors. Following the modeling, since power system is a large-scale and highly nonlinear dynamic system, power flow calculation involves solving higher-dimensional sparse nonlinear algorithmic equations based on nodal admittance form. Load flow equations allow us to compute bus voltage magnitudes and phase angles as well as branch current magnitudes. Solving nonlinear equations is related to iterative computations, which are data- and computation-intensive.

SSSA can be replaced by a series of load flow analyses. However, the computation would be more intensive if a rigorous load flow calculation is used. Some studies are focusing on SSSA to improve the performance [2, 3]. Many highly simplified algorithms are worked out to save iterative computations, and a few approaches with parallel load flow analysis are proposed. *Graphics processing unit* (GPU) is becoming an attractive accelerator for parallel computation [4] and can achieve high performance for general-purpose computation. Load flow analysis is intended to solve high-dimensional sparse nonlinear equations, and sensitivity analysis solves changes in network state based on the load flow results, which uses a number of sparse matrix operations, such as matrix multiplication, addition, and inversion. These features make GPU a suitable and viable solution for SSSA. In this study, while we have provided a GPU-based SSSA solution [5], in practice, a considerable amount of work is needed to continue improving its speed and performance.

Many operations in SSSA can be parallelized on GPU to improve the performance. However a hybrid approach is necessary to combine CPU and GPU computations in GPU-based SSSA. Branch prediction or speculative execution is not supported on GPU, which is not good at iterative operations and judgements of convergence in solving nonlinear equations. For a large sparse matrix, the nonzero storage and computation mechanism should be adopted. The proposed method can combine various small matrices into one matrix in multiplication to use the GPU threads better.

The reminder of this paper is organized as follows: the background and related work are described in Section 2. Section 3 describes the workflow of static state security analysis and its modules. Section 4 addresses the optimization of matrix operations. Section 5 evaluates the performance of the proposed system and addresses optimization issues. Section 6 contains the conclusions of this study as well as a discussion of future improvements in the proposed system.

2. Related Work

SSSA is an important computation tool in the design and operation of large, interconnected power systems. It determines whether a system is operating in a secure state at any given time with respect to unforeseen outages m in N buses, lines, transformers, or generators. $N - 1$ branch outage simulation is a basic validation of a power grid safety [6]. SSSA can be implemented by running one load flow calculation for each outage case. If a large number of cases are provided, the time needed for calculation and analysis can be very long. Hence, other methods, such as DC *(Direct Current)* power method and sensitivity analysis method, have been proposed to speed up static state security analysis.

The DC power method [7] can solve nonlinear equations in power systems by transforming them into linear equations. This reduces computational complexity to make calculations more efficient but can lead to poor precision. The DC power method can only check for the overload in practical application. If the voltage is raised above the upper limit, it will not be checked. The DC power method is commonly used to design power systems, for which the active power flows are an important concern.

In sensitivity analysis, the offline of one transmission line is regarded as a perturbation under normal circumstances [8]. The method involves calculating a sensitivity matrix from the Taylor series expansion of load flow equations. The impact of line outage can be simulated by net injection and withdraw changes. Many analyses of outages on the same network conduct only one load flow calculation and perform their sensitivity analysis on the load flow results. This bypasses many intermediate, iterative steps of the calculation and significantly increases the efficiency of line-off analysis. This method can not only improve performance and precision, but also obtain active power, reactive power, voltage magnitude, and angle at the system node. Therefore, it is commonly used in general practice.

Static state security analysis pays more attention to the calculation of the linear algebraic operation. Reference [9] presents a systematic approach for a large set of frequently encountered dense linear algebra operations, but it is shown to yield new high-performance algorithms. There are many researches based on GPU, in which cuBLAS [10] is the most commonly used. cuBLAS is a linear algebraic library on GPU, which is an implementation of BLAS *(Basic Linear Algebra Subprograms)* on NVIDIA's CUDA *(Compute Unified Device Architecture)* runtime. When programming on cuBLAS, we use it like BLAS and do not care about the details of thread modeling or the storage model of CUDA programming.

Reference [11] proposed a hybrid CPU-GPU implementation of sparse Cholesky factorization based on the multifrontal method. Reference [12] introduced an efficient GPU-based sparse solver for circuit problems and developed a hybrid parallel LU factorization approach combining task-level and data-level parallelism on GPUs. Reference [13] proposed an implementation of the *Newton-Raphson (NR)* load flow algorithm, as it pertained to parallelizing and implementing in CUDA. However, there is no simple and efficient static state security analysis method based on GPU to conduct $N - 1$ branch outage simulations.

3. Workflow and Modules of Static State Security Analysis

3.1. Overview of Static State Security Analysis. Sensitivity analysis is a method to be highly parallelized, which is convenient to be used with GPU to accelerate calculations in static state security analysis. The topology of the power grid can be simply modeled as a graph with m nodes and n edges, in which the nodes represent buses (power stations or transformers) and the edges represent transmission lines [14]. The graph network can be converted into a nodal admittance matrix or a nodal impedance matrix. The two matrices are highly sparse for a large power system. According to the conservation of complex power theorem, the load flow equations can be written as follows:

$$\Delta P_i = P_{is} - V_i \sum_{j \in i} V_j \left(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right) = 0,$$

$$\Delta Q_i = Q_{is} - V_i \sum_{j \in i} V_j \left(G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right) = 0, \quad (i = 1, 2, \ldots, n), \tag{1}$$

where P_{is} and Q_{is} are the injected active and the reactive powers, respectively, at node i. G_{ij}, B_{ij} are the elements of nodal admittance matrix. Load flow calculations can be roughly considered as the problem of solving node voltages V_i, θ_i $(i = 1, 2, \ldots, n)$ at each node when the injecting complex powers P_{is}, Q_{is} $(i = 1, 2, \ldots, n)$ are given. This nonlinear equation can be solved by the *Newton-Raphson* (NR) method, which is based on a shortened Taylor series. NR needs to iteratively solve (2) until it converges within a given tolerance of ΔP_i, ΔQ_i $(i = 1, 2, \ldots, n)$:

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = J \begin{bmatrix} \Delta \theta \\ \dfrac{\Delta V}{V} \end{bmatrix}. \tag{2}$$

For a large system, the load flow calculation may require significant computational resources to calculate, store, and factorize the Jacobian matrix J. Following load flow analysis, the normal state of the system is determined. If there is a randomly injected power disturbance ΔW or network change ΔY, the state vector is correspondingly changed by ΔX. The equation can be expressed as follows:

$$W_0 + \Delta W = f\left(X_0 + \Delta X, Y_0 + \Delta Y\right) \tag{3}$$

in which W_0 is the active and reactive power of nodes in normal state, X_0 is the voltage vector of the nodes, and Y_0 is the normal network parameter. When power disturbance is ignored, $\Delta W = 0$, and Taylor series expansion [15] is used for (3). Finally, we obtain the following equations:

$$\begin{aligned} \Delta x &= S_0 \Delta W_y, \\ \Delta W_y &= \left[I + L_0 S_0 \right]^{-1} \cdot \left(-f_y' \left(X_0, Y_0 \right) \Delta Y \right). \end{aligned} \tag{4}$$

In the above, S_0 is the sensitivity matrix, which is equal to J^{-1}. If only one line-off is considered in SSSA, the injected power change ΔW_y of nodes along the offline can be obtained by

$$\begin{bmatrix} \Delta P_i \\ \Delta Q_i \\ \Delta P_j \\ \Delta Q_j \end{bmatrix} = H^{-1} \begin{bmatrix} P_{ij} \\ Q_{ij} \\ P_{ji} \\ Q_{ji} \end{bmatrix}, \tag{5}$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$+ \begin{bmatrix} -H_{ij} & 2P_{ij} - N_{ij} & H_{ij} & N_{ij} \\ -J_{ij} & 2Q_{ij} - L_{ij} & J_{ij} & L_{ij} \\ H_{ji} & N_{ji} & -H_{ji} & 2P_{ji} - N_{ji} \\ J_{ji} & L_{ji} & -J_{ji} & 2Q_{ji} - L_{ji} \end{bmatrix}$$

$$\times \begin{bmatrix} S_{ii}^{(1)} & S_{ii}^{(2)} & S_{ij}^{(1)} & S_{ij}^{(2)} \\ S_{ii}^{(3)} & S_{ii}^{(4)} & S_{ij}^{(3)} & S_{ij}^{(4)} \\ S_{ji}^{(1)} & S_{ji}^{(2)} & S_{jj}^{(1)} & S_{jj}^{(2)} \\ S_{ji}^{(3)} & S_{ji}^{(4)} & S_{jj}^{(3)} & S_{jj}^{(4)} \end{bmatrix}, \tag{6}$$

$$H = I + L \cdot S. \tag{7}$$

In the above equations, H_{ij}, N_{ij}, J_{ij}, and L_{ij}, the elements of the Jacobian matrix J, are calculated as follows:

$$\begin{aligned} H_{ij} &= \frac{\partial P_i}{\partial \theta_j} = V_i V_j \left(G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right), \\ N_{ij} &= V_j \frac{\partial P_i}{\partial V_j} = V_i V_j \left(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right), \\ J_{ij} &= \frac{\partial Q_i}{\partial \theta_j} = -V_i V_j \left(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right), \\ L_{ij} &= V_j \frac{\partial Q_i}{\partial V_j} = V_i V_j \left(G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right), \quad (j \neq i) \end{aligned} \tag{8}$$

and $S_{ij}^{(1)}$, $S_{ij}^{(2)}$, $S_{ij}^{(3)}$, and $S_{ij}^{(4)}$, the elements related to the offline node in the sensitivity matrix, are calculated as follows:

$$\begin{aligned} S_{ij}^{(1)} &= \frac{\partial \theta_i}{\partial P_j}, \\ S_{ij}^{(2)} &= \frac{\partial \theta_i}{\partial Q_j}, \\ S_{ij}^{(3)} &= \frac{1}{V_i} \frac{\partial V_i}{\partial P_j}, \\ S_{ij}^{(4)} &= \frac{1}{V_i} \frac{\partial V_i}{\partial Q_j}. \end{aligned} \tag{9}$$

We find that the items on the right-hand side of (6) come from load flow calculation, and only two 4×4 matrix operations occur in the equation.

The overall workflow of the static state security analysis system is shown in Figure 1. It consists of four main modules: I/O, preprocessing, power flow calculation, and static state security analysis.

When the original data is input, it is preprocessed first for power flow calculations. The system then conducts a full power flow calculation and determines the Jacobian matrix J, which is prepared for static state security analysis. Finally, the

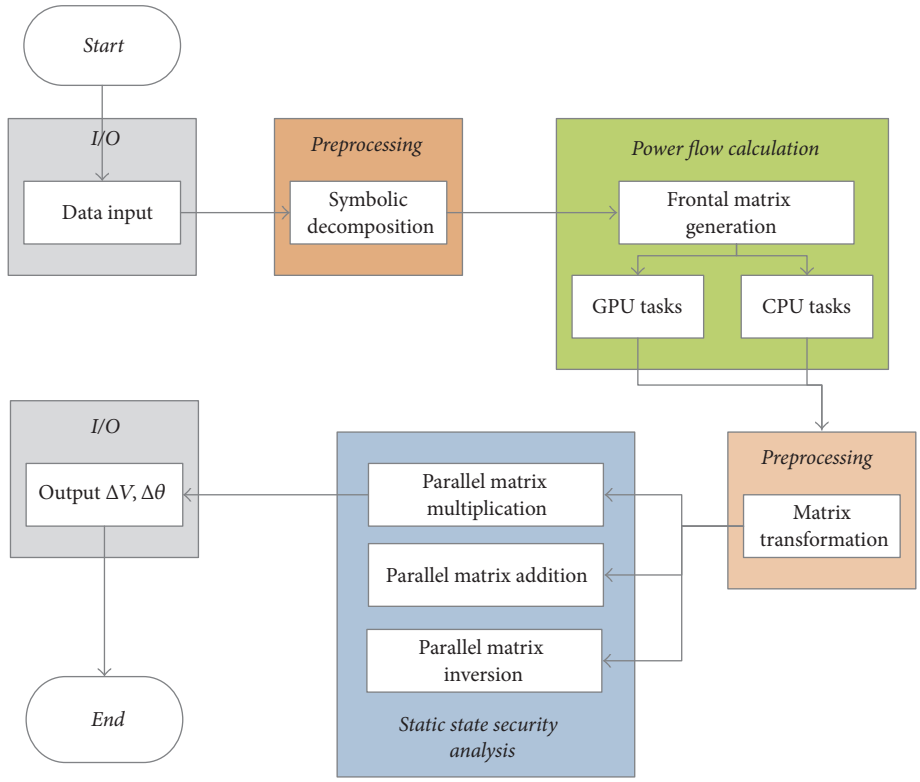


Figure 1: The overall workflow of static state security analysis.

changing network parameters are assumed by a line-off and new system state variables identified. Once a set of critical elements are simulated to be tripped, we can determine whether there are equipment-limit violations through SSSA. Hence, the state of power system (secure or unsecured) can be easily determined.

3.2. Preprocessing. Data preprocessing is a major and essential stage in power flow calculation, which can speed up static state security analysis. There are three processings: sparse matrix storage, node numbering optimization, and result storage optimization, which aim at reducing storage space for matrices and speeding up the calculation of nonzero elements.

Following raw data input, data in large sparse matrix is compressed in CSR *(Compressed Row Storage)* format [16]. CSR is very efficient due to an indirect addressing step for every single scalar operation in matrix vector. The subsequent nonzero elements in matrix rows are placed in contiguous locations in memory and traversed in a row-wise fashion.

By matrix sorting with the minimum-degree minimum-length algorithm [17] or the minimum-degree minimum-number algorithm [18], the rows and columns of sparse matrix are moved to avoid creating new nonzero elements in the matrix during factorization, and the forward and backward substitutions can save many nonzero computations. Such node numbering optimizations are conducted with an elimination tree, which can determine the position of injection elements in factorization.

Static state security analysis is performed on the results of power flow calculations. Since three read operations are necessary when accessing CSR-formatted data every time, it incurs a considerable costs for I/O operations. Hence, the matrix is decompressed prior to calculation. It may take up a lot of space to assemble several matrices, and the memory may not be large enough to store the matrices. Therefore, preprocessing involves the following three steps:

(1) Analyze the result of power flow calculation J, where sensitivity matrix is obtained by the inverse S_0. Allocate memory to store matrices assembled from S_0.

(2) Generate submatrices and vectors from S_0 with L and S in (6) and (7), respectively.

(3) Check matrix size. If it does not exceed the capacity, copy it directly into GPU device memory; otherwise, partition the matrix and copy submatrices into device.

3.3. Power Flow Calculation. Power flow calculation is executed with the *Newton-Raphson* (NR) method to solve large sparse nonlinear equations. The NR method repeatedly solves large, sparse, linear systems of equations. It is a significant challenge of computational and memory capacity to factorize the Jacobian matrix at each iteration; hence, it is crucial to design a fast and efficient solution for the factorization. The multifrontal algorithm is a particularly useful method for factoring large sparse linear systems and is adopted in the proposed solution. The workflow of the method is shown in Figure 2.

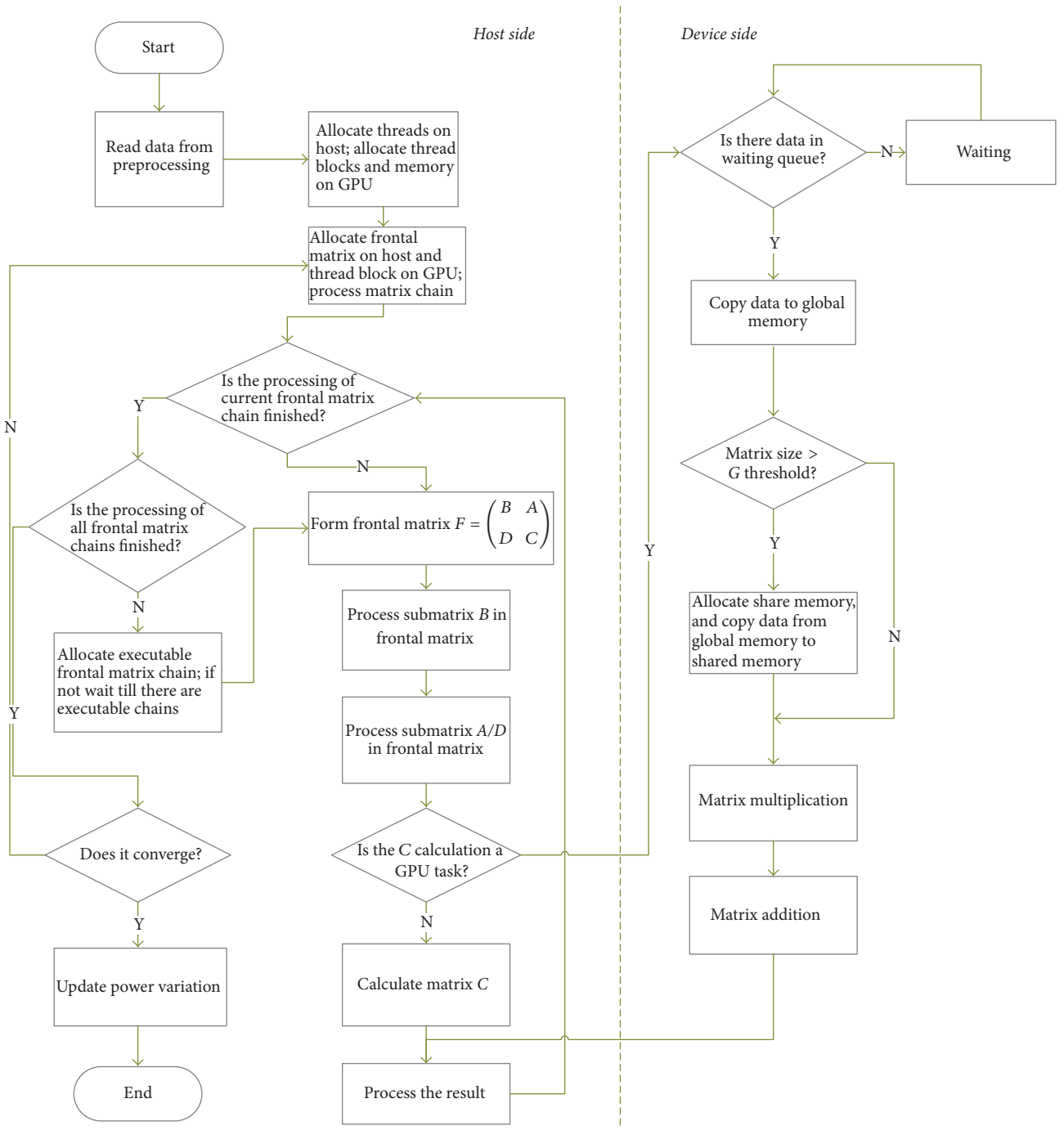
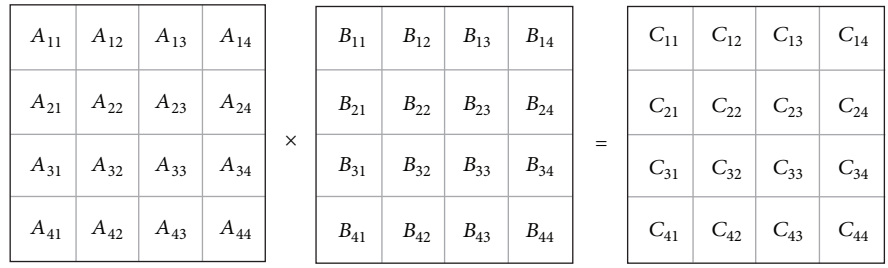


Figure 2: The workflow of power flow calculation.

The multifrontal algorithm [19] is an effective method to solve large sparse matrices operation. A frontal matrix is a small, dense matrix. The term "*multi*" here refers to the fact that multiple frontal matrices are used during the factorization. The multifrontal method turns the factorization of sparse matrix into a sequence of factorizations of smaller, dense matrices organized as an elimination tree or an assembly tree. The method can get satisfactory data locality and great potential for parallelization. For example, [20] investigates an automatic tuning of SpMV (Sparse Matrix Vector) multiplication kernel in a partitioned global address space language, which supports a hybrid thread- and process-based communication layer for multicore systems. The multifrontal method proposed here is associated with super-nodal implementation, so that multiple columns with the same nonzero patterns can be grouped together as a dense kernel for concurrent factorization.

The method is formulated in terms of frontal matrices and update matrices. The processing order of frontal matrices is determined by the elimination tree from the preprocessing.



Thread 1: $C_{11} = A_{11}B_{11} + A_{12}B_{21} + A_{13}B_{31} + A_{14}B_{41}$
Thread 2: $C_{12} = A_{11}B_{12} + A_{12}B_{22} + A_{13}B_{32} + A_{14}B_{42}$
$\vdots$
Thread 16: $C_{44} = A_{41}B_{14} + A_{42}B_{24} + A_{43}B_{34} + A_{44}B_{44}$

Figure 3: Submatrix multiplication with 16 threads.

The order is expressed by a frontal matrix chain, in which each frontal matrix is designated as a leaf and processed by a CPU thread. For node j, subtree update matrix $\overline{U}_j$ is the sum of all subtree update matrices; frontal matrix F_j is formed by assembly over $\overline{U}_j$ and can be partitioned into four submatrices A, B, C, and D with (10). Then the Cholesky factor vector and the update matrix of j can be solved with following Cholesky factorization:

$$\begin{pmatrix} B & A \\ D & C \end{pmatrix} = \begin{pmatrix} L_1 & 0 \\ L_2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & C - L_2U_2 \end{pmatrix} \begin{pmatrix} U_1 & U_2 \\ 0 & 1 \end{pmatrix}. \quad (10)$$

It will take a considerable amount of time for matrix operations in the multifrontal method. A threshold is defined to distinguish CPU from GPU tasks and is derived from matrix calculations. If the number of the computations is greater than the threshold, the main thread will allocate tasks to GPU. Otherwise, a small-scale matrix is computed on CPU. Nodes in frontal matrix chain are calculated one by one until all of them are computed. ΔX is then computed backward through substitute, decomposed matrices L and U to update X. If the absolute value of ΔX is less than 10^{-8}, which is regarded as convergence, the calculation is concluded; otherwise, the iterations will continue to get a convergent result.

3.4. Static State Security Analysis. In this module, the GPU-based sensitivity method is used for static state security analysis. The elements in the matrices of (6) are prepared from the previous load flow analysis. It consists of four steps.

Step 1. Calculate $T = L \cdot S$ in (6) by matrix multiplication.

Step 2. Calculate $H = T + I$ in (6) by matrix addition.

Step 3. Calculate H^{-1} by matrix inversion from H.

Step 4. Calculate (5) by using a matrix multiplying vector.

Two matrix multiplication and one matrix inversion operations are involved during the analysis. Following these steps, the changes in the node state variables can be obtained, so that the power flow on each branch can be acquired following line outage. Moreover, many cases of SSSA can be combined in parallel on a GPU to improve the efficiency.

4. Optimization of Matrix Operations

4.1. Small Matrix Multiplication. In 4×4 matrix multiplication on GPU, 16 threads are executed for concurrent element multiplication in one block, shown in Figure 3. The 16 threads are completely independent and communicate with one another in the block. However, the same instruction is executed in a warp, which is allowed to launch 32 threads at once. Whether there are 32 or 16 threads in a task, they are launched by a warp with the same time cost. Hence if only one submatrix multiplication occurs in a block with only a warp, 16 threads are launched and the other 16 are idle.

It is a best practice to allocate more than one multiplication operation to a block. A block can contain a maximum of 1024 threads, which can store 64 submatrix multiplication operations. Host threads can combine their data into a large matrix and copy it to share the memory of the block. Since the instructions are the same in a block, it is necessary to access own data of the large matrix in each thread. The distribution of logical memory is shown in Figure 4. We assume that every 16 threads process a matrix multiplication operation and 64 matrix multiplications can be concurrently handled in a block.

4.2. Small Matrix Inversion. Between J and S_0, there is a matrix inversion operation. In general, there are two methods for matrix inversion. One is the adjoint matrix method, which executes the calculation of $A^{-1} = A^*/|A|$. The adjoint matrix A^* of A must be calculated first for this. It is too complicated to derive the expression of A^* for matrix A of order greater than 3. The other matrix inversion method is the elementary transformation method, which is simpler than the adjoint matrix method for high-order matrices, but more iterations are required and inefficient in terms of GPU

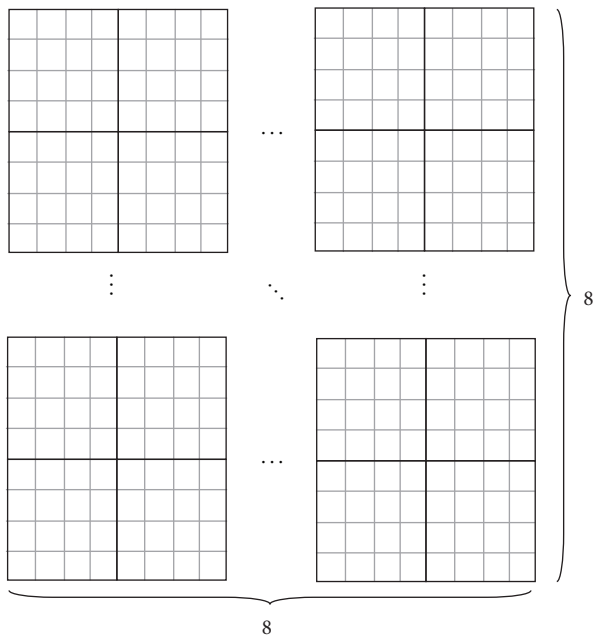


Figure 4: The distribution of logical memory.

resource consumption. In order to enhance efficiency, (11) is used to partition matrix inversion:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}B\left(D - CA^{-1}B\right)^{-1}CA^{-1} & -A^{-1}B\left(D - CA^{-1}B\right)^{-1} \\ D\left(D - CA^{-1}B\right)^{-1}CA^{-1} & \left(D - CA^{-1}B\right)^{-1} \end{bmatrix}. \tag{11}$$

If a 4×4 matrix is provided, it can be divided into four 2×2 matrices. In this manner, the inversion task is divided and assigned to four threads, which can complete their tasks almost simultaneously. In (11), only the operations of matrix multiplication, addition, and 2×2 matrix inversion are given. 2×2 matrix inversion can be expressed as follows:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \begin{bmatrix} \dfrac{d}{ad - bc} & \dfrac{-b}{ad - bc} \\ \dfrac{-c}{ad - bc} & \dfrac{a}{ad - bc} \end{bmatrix}. \tag{12}$$

In this way, four threads can be used with (12) to calculate the inversion of A, B, C, and D in formula (11). The results are substituted into (11) to work out the inversion of the 4×4 matrix.

4.3. Vector Multiplication by Cross-Combining Storage. When solving (5), each change in power flow is obtained by multiplying 4×1 row vector by 1×4 column vector. A set of vector-vector multiplication can be combined into a matrix vector-vector multiplication as Figure 5.

In sensitivity analysis, $32n$ (n is a positive integer) row vectors are merged into one long row vector placed in global memory. Global memory allows a warp of 32 threads to access it concurrently. As Figure 6 shows, there are 32 4×1 row

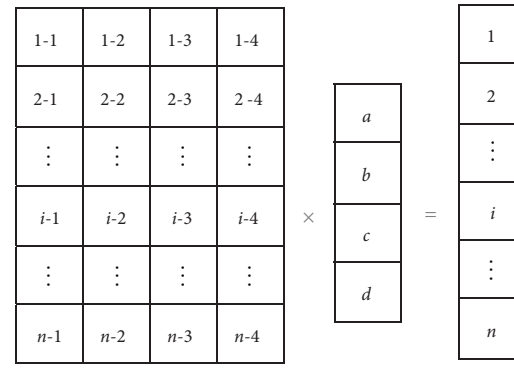


Figure 5: Matrix vector multiplication.

vectors; one long row vector x is created in global memory. The first, second, third, and fourth column elements of the 32 row vectors are set to $x[0 : 31]$, $x[32 : 63]$, $x[64 : 95]$, and $x[96 : 127]$, respectively. When row vector x is multiplied by column vector y, $x[0 : 31]$, $x[32 : 63]$, $x[64 : 95]$, and $x[96 : 127]$ are multiplied by $y[a]$, $y[b]$, $y[c]$, and $y[d]$, respectively. Since vector y is accessed many times, y is stored in the shared memory.

A number of row vectors are stored in one row vector by cross-combination. It is necessary to execute a reduce operation after vector multiplication. The result of vector multiplication is stored back into x. The four result elements $(i, i + 32, i + 64, i + 96)$ are grouped, summed up, and stored at i. Eventually, $x[0–32]$ is the result of 32 multiplications of a 1×4 vector by another 4×1 vector.

4.4. RC-MM Storage Method. In general practice, cublasSgemm function in cuBLAS is called for GPU matrix multiplication. The matrices in C/C++ are in row-major order, but cuBLAS assumes that the matrices are stored in column-major order in the devices. The order exchange is a time-consuming operation. Hence the RC-MM *(Row Column-Matrix Multiplication)* method is adopted, and A, B, and C are stored in column-major, row-major, and column-major order, respectively, in $C = AB$.

For the multiplication in the multifrontal method, one matrix is stored in row-major order and the another matrix in column-major order. The matrices of the same frontal chain share an array, because of which the entire row or column data is copied to global memory. Each computation only uses part of the data, which avoids having to move large amounts of data on demand.

5. Performance Evaluation

5.1. Dataset and Experimental Environment. The experiments are conducted on a server with an Intel i7 950 (3.07 GHz) CPU, 16 GB memory, and NVIDIA GeForce GTX460. The CentOS 5.9 Linux operation system with CUDA 4.0 is used. From MATPOWER [21], a professional software in power calculation derived from a real power grid and some IEEE standard testing datasets are chosen, shown in Table 1.

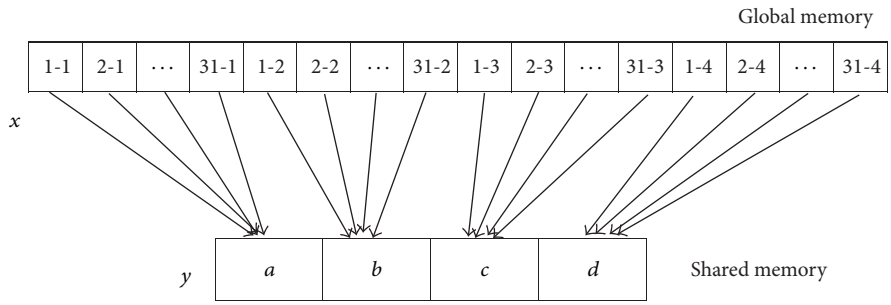


Figure 6: Matrix vector multiplication by cross-combining storage.

Table 1: Test datasets for SSSA experiments.

Dataset name	CA300	CA3012wp	CA3120sp	CA5472	CA5492wp	CA6024	CA6240
Number of nodes	300	3012	3120	5472	5492	6024	6240
Number of lines	411	3572	3693	9794	9824	10706	11069

CA300 is the power flow data for the IEEE 300 bus test case. CA3012wp and CA3120sp are the power flow data cases for the Polish system winter 2007-08 evening peak and summer 2008 morning peak, respectively. In order to evaluate the computation involved in a large power system, two identical Polish power systems are connected to form the large symmetric power system CA5472, CA5492wp, CA6024, and CA6240.

The speedup ratio S is defined in (13), where $T_{\text{CPU\_time}}$ and $T_{\text{GPU\_time}}$ are the execution times on CPU and GPU, respectively. The program executed on CPU is a multithread program on multicore CPU.

$$S = \frac{T_{\text{CPU\_time}}}{T_{\text{GPU\_time}}} \times 100\%. \tag{13}$$

5.2. Evaluation of Static State Security Analysis System. Executed on GPU and CPU platforms, the datasets in Table 1 are used for testing. The results are shown in Table 2.

In Table 2, there is a 4-core processor in the CPU, with which the program can almost achieve best performance. But with increasing of the number of active processor cores in the compute node, the program on the CPU platform suffers serious performance degradation. In dataset CA3012wp, the calculation time on an 8-core CPU increased by 13% compared with a 4-core CPU.

The experimental results show that the execution time on GPU is shorter than on CPU, except on a scale of 300 nodes. It takes much time to transfer data between host and device in GPU. When the scale is 300 nodes, the transfer time cannot be ignored, and it is occupied by a significant proportion of the execution time on GPU. On the other hand, for only 300 submatrices, every two submatrices are processed in a warp on average. However for 336 cores on GTX 460, it means that over half the cores are idle throughout the processing, which is a serious waste of GPU resources to hinder overall performance.

Excluding the dataset of the 300 nodes, the accelerating effect of GPU computation is reflected, and significant speedup is revealed with increasing data scale. GPU computing can reduce calculation time by 40% compared with the execution on 4-core CPU. The increasing trend of GPU speedup is shown in Figure 7.

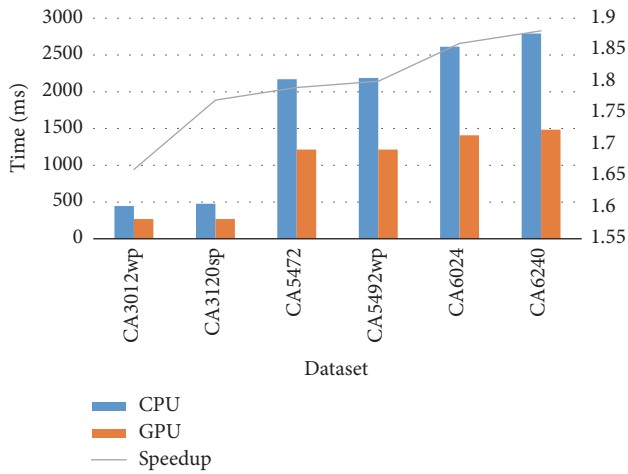


Figure 7: Comparison in terms of execution time between CPU and GPU speedup.

5.3. Evaluation of Small Matrix Multiplication. Equation (6) is chosen to run the performance experiments. The formula consists of 4×4 matrix multiplication and 4×4 matrix addition. Two experiments are done. In Experiment 1 (Exp1), two combined submatrices are stored in one block after optimization. In Experiment 2 (Exp2), one submatrix is stored in one block. For different matrix storages, the experimental results for matrix multiplication and addition in (6) are shown in Figure 8.

The optimization method can achieve a speedup of approximately 1.7. A number of parallel threads are launched by a warp. If the matrices are not merged into a block, half the threads will be idle in a warp, which can launch 32 threads.

Table 2: Comparison in terms of execution time between CPU and GPU.

Scale	200	3000		5000		6000	
Dataset name	CA300	CA3012wp	CA3120sp	CA5472	CA5492wp	CA6024	CA6240
CPU (ms)	10.0	446.6	476.7	2170.0	2186.7	2603.3	2793.3
GPU (ms)	19.7	269.3	269.6	1214.3	1213.0	1408.0	1483.7

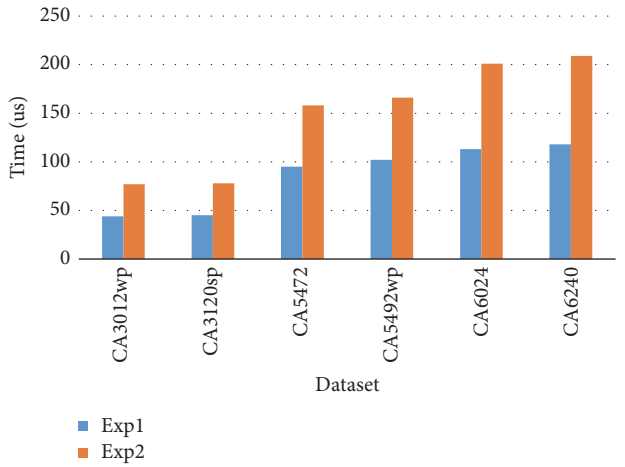


Figure 8: Comparison in terms of matrix multiplication execution times of the two experiments.

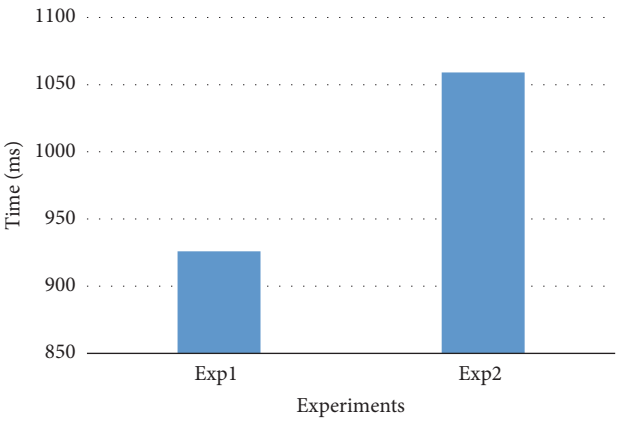


Figure 9: Comparison in terms of small matrix inversion times of the two experiments.

Following matrix combination, the 32 threads in the warp can be kept busy, which indicates that the optimization method can exhibit good performance.

5.4. Evaluation of Small Matrix Inversion. Two experiments are conducted on 4×4 matrix inversion in (5). cuBLAS has provided a large matrix inversion function with low efficiency for batch small matrices inversion. The small batch matrix inversion method is implemented on GPU kernel, in which one submatrix is processed by one thread. The small batch matrix inversion function is called in Exp1 once, and 100 inversion threads are scheduled on the GPU. The cuBLAS matrix inversion function is called 100 times in Exp2. The results show that the small matrix inversion method yields better performance (Exp1 925 ms versus Exp2 1055 ms) in Figure 9.

6. Conclusion and Future Work

In this paper, GPU-based static state security analysis is proposed for power systems. The GPU-based multifrontal method is implemented to solve a large sparse matrix, and sensitivity analysis is chosen for static state security analysis on GPU. To make full use of GPU device, several optimization methods of matrix operations are presented, such as data combination in multiple small-scale matrix multiplication operations and the partition matrix method for matrix inversion.

Experimental results indicate that the proposed algorithm on GPU can significantly improve system performance. Our results show a speedup of 1.7–1.9 with power system simulation cases from a scale of 3,000 to 6,000.

In future work, it may be desirable to further improve performance that the system and methods could be ported to more scalable distributed memory environment, such as multi-GPUs [22]. We can also use the compilers to speed up system migration and dynamic task scheduling in CPU-GPU heterogeneous parallel systems. Our way of dealing with small-scale matrices can be used in scientific calculations in other fields, and the processing method of special-dimensional matrix can be extended to all-dimensional matrices.

Competing Interests

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 61133008), the National 973 Key Basic Research Plan of China (no. 2013CB2282036), Major Subject of State Grid Corporation of China (no. SGCC-MPLG001(001-031)-2012), the National 863 Research and Development Program of China (no. 2011AA05A118), and the National Science and Technology Pillar Program (no. 2012BAH14F02).

References

[1] J. D. Glover, M. S. Sarma, and T. Overbye, *Power System Analysis and Design*, Cengage Learning, 2016.

[2] P. Ding, Y. Li, D. Xu, F. Tian, J. Yan, and Z. Yu, "Improved algorithm of fast static state security analysis of power systems," *Proceedings of the Chinese Society of Electrical Engineering*, vol. 30, no. 31, pp. 77–82, 2010.

[3] G. Zhou, X. Zhang, Y. Lang et al., "A novel GPU-accelerated strategy for contingency screening of static security analysis," *International Journal of Electrical Power & Energy Systems*, vol. 83, pp. 33–39, 2016.

[4] J. D. Owens, D. Luebke, N. Govindaraju et al., "A survey of general-purpose computation on graphics hardware," *Computer Graphics Forum*, vol. 26, no. 1, pp. 80–113, 2007.

[5] Y. Chen, H. Jin, H. Jiang, D. Xu, R. Zheng, and H. Liu, "GPU-based static state security analysis in power systems," in *Proceedings of the 9th Asia-Pacific Services Computing Conference (APSCC '15)*, pp. 258–267, Bangkok, Thailand, December 2015.

[6] X. Li and Z. Guo, "The transmission interface real power flow control based on N-1 static safety restriction," *Electric Power*, vol. 38, no. 3, pp. 26–28, 2005.

[7] K. Purchala, L. Meeus, D. Van Dommelen, and R. Belmans, "Usefulness of DC power flow for active power flow analysis," in *Proceedings of the IEEE Power Engineering Society General Meeting*, pp. 454–459, San Francisco, Calif, USA, June 2005.

[8] X. Wang, W. Fang, and Z. Du, *Modern Power System Analysis*, Science Press, 2016.

[9] P. Bientinesi, J. A. Gunnels, M. E. Myers, E. S. Quintanaorti, and R. A. van de Geijn, "The science of deriving dense linear algebra algorithms," *ACM Transactions on Mathematical Software*, vol. 31, no. 1, pp. 1–26, 2005.

[10] NVIDIA, "cuBLAS," https://developer.nvidia.com/cublas.

[11] R. Zheng, W. Wang, H. Jin, S. Wu, Y. Chen, and H. Jiang, "GPU-based multifrontal optimizing method in sparse Cholesky factorization," in *Proceedings of the 26th IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP '15)*, pp. 90–97, IEEE, Ontario, Canada, July 2015.

[12] X. Chen, L. Ren, Y. Wang, and H. Yang, "GPU-accelerated sparse LU factorization for circuit simulation with performance modeling," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 3, pp. 786–795, 2015.

[13] D. J. Sooknanan and A. Joshi, "GPU computing using CUDA in the deployment of smart grids," in *Proceedings of the SAI Computing Conference (SAI '16)*, pp. 1260–1266, London, United Kingdom, July 2016.

[14] L. Xuan, L. Tianqi, and L. Xingyuan, "A novel evolving model for power grids," *Science China Technological Sciences*, vol. 53, no. 10, pp. 2862–2866, 2010.

[15] S. Wang, Z. Zheng, and C. Wang, "Power system transient stability simulation under uncertainty based on Taylor model arithmetic," *Frontiers of Electrical and Electronic Engineering in China*, vol. 4, no. 2, pp. 220–226, 2009.

[16] J. L. Greathouse and M. Daga, "Efficient sparse matrix-vector multiplication on GPUs Using the CSR storage format," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '14)*, pp. 769–780, November 2014.

[17] A. Gómez and L. G. Franquelo, "An efficient ordering algorithm to improve sparse vector methods," *IEEE Transactions on Power Systems*, vol. 3, no. 4, pp. 1538–1544, 1988.

[18] R. Betancourt, "An efficient heuristic ordering algorithm for partial matrix refactorization," *IEEE Transactions on Power Systems*, vol. 3, no. 3, pp. 1181–1187, 1988.

[19] J. W. Liu, "The multifrontal method for sparse matrix solution: theory and practice," *SIAM Review*, vol. 34, no. 1, pp. 82–109, 1992.

[20] S. G. Li, C. J. Hu, J. C. Zhang, and Y. Q. Zhang, "Automatic tuning of sparse matrix-vector multiplication on multicore clusters," *Science China Information Sciences*, vol. 58, no. 9, pp. 1–14, 2015.

[21] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, 2011.

[22] D. Chen, W. Chen, and W. Zheng, "CUDA-Zero: a framework for porting shared memory GPU applications to multi-GPUs," *Science China Information Sciences*, vol. 55, no. 3, pp. 663–676, 2012.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 1360413, 12 pages
https://doi.org/10.1155/2017/1360413

Research Article

RAID-6Plus: A Comprised Methodology for Extending RAID-6 Codes

Ming-Zhu Deng, Nong Xiao, Song-Ping Yu, Fang Liu, Lingyu Zhu, and Zhi-Guang Chen

State Key Laboratory of High Performance Computing, College of Computer, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Ming-Zhu Deng; dk_nudt@126.com

Received 23 September 2016; Revised 26 December 2016; Accepted 10 January 2017; Published 23 February 2017

Academic Editor: Laurence T. Yang

Existing RAID-6 code extensions assume that failures are independent and instantaneous, overlooking the underlying mechanism of multifailure occurrences. Also, the effect of reconstruction window is ignored. Additionally, these coding extensions have not been adapted to occurrence patterns of failure in real-world applications. As a result, the third parity drive is set to handle the triple-failure scenario; however, the lower level failure situations have been left unattended. Therefore, a new methodology of extending RAID-6 codes named RAID-6Plus with better compromise has been studied in this paper. RAID-6Plus (Deng et al., 2015) employs short combinations which can greatly reuse overlapped elements during reconstruction to remake the third parity drive. A sample extension code called RDP+ is given based on RDP. Moreover, we extended the study to present another extension example called X-code+ which has better update penalty and load balance. The analysis shows that RAID-6Plus is a balanced tradeoff of reliability, performance, and practicality. For instance, RDP+ could achieve speedups as high as 33.4% in comparison to the RTP with conventional rebuild, 11.9% in comparison to RTP with the optimal rebuild, 47.7% in comparison to STAR with conventional rebuild, and 26.2% for a single failure rebuild.

1. Introduction

In modern data centers, RAID-6 credited for performance and reliability are among the most popular configurations to be deployed. However, more devices, larger disks, unimproved reliability, increased bit errors, and less-reliable hardware all expose modern storage systems to higher vulnerability, demanding RAID-6 evolvement with higher and more flexible reliability care [1]. Thus, the extension of the existing RAID-6 coding scheme is worth attention.

In fact, there are various RAID-6 coding algorithms available to be extended for higher failure tolerance. For example, RS codes [2] can be applied with various parameters while EVENODD [3] and RDP [4] are XOR-based for faster computation [5]. Regarding higher reliability, many attempts have been made by extending the existing RAID-6 codes. Blaum et al. generalized EVENODD for arbitrary failure scenarios [6]. Huang and Xu proposed the STAR code [7] to protect a storage array from triple failures. In fact, the STAR code is another extension of EVENODD. Goel and Corbett extend RDP to RTP [8] in 2012 to provide a faster coding algorithm for triple failures. Similarly, the Triple-STAR code [9] is extended from Rotary-code [10] by adding a more diagonal parity column to tolerate triple failures.

Codes along this direction can provide satisfactory solutions to triple failures, but the following problems still remain unsolved especially the lack of flexibility for enabling higher reliability:

(i) Existing codes assume that failures are independent, instantaneous, and occurrences of failures conform to the exponential distribution [11, 12]. This ideal assumption does not apply to the fault pattern of modern storage systems [13]. Furthermore, these codes are not designed to support multifailure degradations; such degradations aim to convert a higher level multifailure into separate low-level multifailures or single failures with a shorter reconstruction window.

(ii) Existing codes largely ignore the pattern of failure occurrences in practice. For example, 99.75% of recoveries are due to single disk failures [14], while

triple whole-disk failures are rare. However, the third parity drive in RTP is set to handle the triple-failure scenario only with single failure rebuild unattended. The third parity drive is then almost wasted.

(iii) Existing codes focus on whole-device level failures while mixed-fault modes are more common in practice [15]. For example, when a fault consisting of two erasures and a sector error occurs, all of the three parity drives must be used. This directly overkills the effects of the third parity drive.

(iv) As throughput is dwarfed by capacity [16], the reconstruction windows have grown exponentially from minutes to hours or even days in practice. This leads to a severe decrease of system performance and poor user experience.

Therefore a new methodology for RAID-6 code extension is in pressing need to support multifailure degradations and a smaller reconstruction window to deliver data reliability in a more flexible manner. In this paper, we propose a compromised code extending methodology with a shorter reconstruction window, named RAID-6Plus [14] to provide higher reliability at the expense of three parity drives.

Existing coding extensions provide absolute reliability for triple failures via full combinations. In contrast, RAID-6Plus employs short combinations which can effectively reuse overlapped elements during reconstruction to remake the third parity drive. This design shortens the reconstruction window of single failures by minimizing the total number of data reads. The possibility of multifailure overlapping in the reconstruction window is therefore significantly diminished. Such features provide RAID-6Plus with (1) a better system performance compared to the RTP and STAR codes and (2) an enhanced reliability compared to the RAID-6. An example extension code called RDP+ is given based on RDP (repetition). Moreover, we expand the study to present another extension example called X-code+ for the sake of vertical codes.

The analysis shows that RAID-6Plus is a balanced compromise among reliability, performance, and practicality. For example, RDP+ achieved least update penalty and far outperforms RTP and STAR both under their optimal reconstruction on encoding and decoding.

The main contributions of this study are as follows:

(i) We developed a new RAID-6 code extending methodology with shorter reconstruction window and lower risk of multifailure with no additional cost incurred compared to RTP and STAR, which provide a balanced tradeoff of flexible reliability and better system performance. This code can be applied to most XOR-based coding schemes and is orthogonal with some previous work on reconstruction speedup [17–20]. They can also be integrated together to further shorten reconstruction window.

(ii) An example extension code called RDP+ is presented based on RDP in terms of encoding and single failure reconstruction improvement.

(iii) Another extension example is given as X-code+ to apply RAID-6Plus with regard to vertical coding schemes. X-code+ shows good performance on single failure rebuild and load balance.

(iv) A new metric called Q-metric is proposed to validate and evaluate the presented extending methodology. Q-metric denotes induced benefit per cost. The higher the Q value, the more competitive and useful the method. Furthermore, the Q-metric is intended to measure the performance improvement per cost, and it is an indicator of the correlation between gains and overheads.

In comparison to the previous work [14, 21], this paper not only provides more details of RDP+ and X-code+, but also reveals the generalized methodology for extending other codes. Also further identification and examination of the problem in the perspective of update penalty and load balance are presented. Additionally, a comprehensive evaluation including both two sample codes together is given. Further, a new metric called Q-metric is proposed to validate and evaluate the proposed extending methodology.

In the remainder of this paper, Section 2 introduces the background and motivation for RAID-6Plus. Section 3 explains the design of RAID-6Plus in detail and presents RDP+ and X-code+. Then, the performance is evaluated in Section 4 and related work is compared in Section 5. Finally, all of the findings of this paper are summarized and concluded in Section 6.

2. Backgrounds and Motivation

In this section, we describe the different failure modes and recovery methods in RAID systems and introduce the concept of multifailure degradation in reconstruction window. This motivates the need to design RAID-6Plus, which exploits multifailure degradation mechanism.

2.1. Failure Modes. Many researches on the massive disks have all shown that (1) mainstream disk drives have device failures or whole-disk failure and sector failures [22]. All these failures directly cause data unavailability. (2) Sector errors are not rare and increase with time [1]. (3) Despite infant mortality, multiple disks tend to fail at a similar age, indicating not only does single failure happen at the device level, but also multiple devices may fail almost simultaneously, calling for higher reliability care [1]. (4) In terms of the correlation between whole-disk failure and sector errors, [1] asserts whole-disk failure can be viewed as the consequence of accumulated sector error and uses the number of reallocated sectors to characterize the probability of whole-disk failure. (5) Further, the longer a functioning device endures, the higher the probability of device failure could be. In other words, other failures could happen in the ongoing process of failure recovery, aggravating system reliability and making it much more vulnerable [23]. In short, the single failure is of vital importance to reduce the window of system vulnerability than RAID-6.

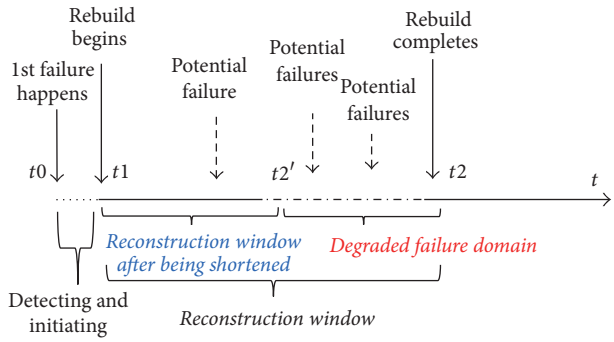


Figure 1: Successive failure happening.

However, real-world statistics show that, among all device failures, single failure accounts for the majority (99.75%), and double failures are not eligible (roughly 8%) and should be taken care of [24]. Meanwhile, often the cases are single failure coupled with other sector failures rather than multi-whole-disk failures [22]. Though there has not been any investigation published on massive SSD, SSD's failure modes must be similar while they differ in some of its vulnerability features, like its inborn limited endurance issues and wearing over time [25].

2.2. Multifailure Degradation. In fact, multiple failures happening at almost exactly the same time is hardly witnessed in real-world systems, which are quite different cases from ideal models for simplification of research. Strictly speaking, multiple failures happen in a successive manner, as shown in Figure 1.

When the first whole-disk failure takes place, the second failure or more will happen mainly in the reconstruction window for the first failure, thus making it a double-failure or multifailure.

It is clear that longer reconstruction window creates more space for multifailures to happen. If we purposely shorten the reconstruction window, the risk of multifailure could be degraded to a lower level, as shown in Figure 1. With proper shortening, a higher level of multifailure is degraded into a lower level of multifailure, alleviating threats to system reliability boundary and data loss [26].

Unfortunately, current codes with higher fault tolerance seldom make use of multifailure degradation mechanism and mainly concentrate on furthering reliability boundary (absolute reliability). They focus on providing inflexible reliability, unable to deliver flexible reliability regarding failure happening mode and probability.

For example, as an extension of the EVENODD code and a modification of the generalized triple-erasure-correcting EVENODD code, STAR code has the same recovery package when single failure happens, showing its inability to contract the reconstruction window [27].

2.3. Motivation. Above all, existing code extensions with higher fault tolerance, like RTP and STAR codes are originally designed for triple whole-device failures. Unfortunately, statistics have shown the probability of single failure overwhelmingly accounts for most while triple whole-device failures are relatively rare, thus making the current RAID-7 system with triple parity drives wasteful. Additionally, mixed-fault modes exhibited in modern storage systems overkill the solution of those codes with higher reliability boundaries. In short, the current RAID-7 system with triple parity coding is unpractical and needs to deliver more flexible reliability [14]. Further, the reconstruction window is exponentially increasing with device capacity, thus worsening user experience and leaving larger space of system vulnerability and data loss. More notably, current coding schemes for triple-failure are unable to shorten reconstruction window.

Therefore, all these factors above motivates extending X-code another way to shorten reconstruction window to provide flexible reliability and make it more practical.

3. Methodology Design

First in this section, the explicit definition and general ideal of the proposed RAID-6Plus is presented. Then RAID-6Plus based on RDP code is instantialized and its construction is illustrated. Further, with regard to vertical codes, some modification has been made and RAID-6Plus has been applied over X-code to get X-code+, which has better load balance.

3.1. Definition and General Ideal. In order to provide solid and flexible reliability against multifailure, RAID-6Plus is defined as a methodology to extend any conventional RAID-6 coding algorithm to delivery extra fault tolerance over double-failure tolerance in a new and practical way. Hereby, the meaning of "plus" is twofold by standing for that higher and more flexible reliability as well as extra cost of an added parity drive compared with RAID-6 configuration after extension. Explicitly, RAID-6Plus keeps the original encoding paradigm of a RAID-6 algorithm to maintain double-failure tolerance and adds one extra redundant drive to aid for accelerating any single failure rebuild scenario by reusing data elements. In order to reuse as many data elements as possible, the optimal reconstruction for single failure rebuild needs to be studied to find out the overlapping elements to be used in the third parity drive encoding. Note that how to find and reuse overlapped data elements would differ among different RAID-6 coding algorithms. In that way, a reasonable compromise is achieved between reliability level and system performance by not only accommodating nonnegligible double-failure but also shortening the reconstruction window to degrade failure and reduce user wait.

In detail, of all the three redundancy drives, the first two are devoted to maintaining a lower bound of reliability, which we denote as "base reliability," given RAID-6 coding scheme is widely deployed in diverse storage system for double-failure concern.

The remaining redundant X drive is dedicated to reducing reconstruction window for single failure of any data drive, which is in charge of user access. Therefore, the key lies in the redundancy coding for X drive. Conventionally, there are three ways for X drive coding: (1) mirroring of a single data disk; (2) short combination; and (3) full combination, which

Table 1: Feature comparison of three coding ways for X drive.

Coding methods	Element example	Involved element	Merit	Shortcoming
Mirroring of single disk	$a1$	1	Simple and can maximize reduction of reconstruction window for specific drive	Only covering replicated drive, not able to cover other drives, causing imbalance and fluctuation
Short combination	$a1 + b1$	2	In between	
Full combination	$a1 + b1 + c1 + d1 + e1$	5	Maximizing fault-tolerance for the whole system	Unable to reduce reconstruction window

is the norm of existing extension methodology. The features of the three coding ways are so obvious that we summarize them in Table 1. Mirroring has a length of only one, suggesting any X element is a replica of some element in other drives. Full combination has the same length of any element in P or Q drive, implying any element in X is the combination of many elements in data drives while short combination is in between.

In order to reduce data reads, short combination with some two elements involved has been employed to encode for X parity. The thought behind it is to find overlapping elements as many as possible on the basis of optimal reconstruction for single data disk in the base code and combine any two of them for the concern of data disk coverage.

In short, those three ways of coding for X parity elements mainly differ in number of elements involved.

3.2. RDP+ over RDP. Since its birth, RDP code has been one of the most popular and efficient RAID-6 codes in academia and industry due to its performance. RTP code extended right from RDP has been proposed years ago. Therefore, another RDP-based code extension is offered by applying the proposed methodology and constructing RDP+.

(i) Optimal Reconstruction for Single Erasure in RDP. In order to explicitly illustrate the construction of RDP+, the optimal reconstruction for single erasure in RDP is provided to get a clear understanding of coding in the third redundant drive.

In RDP, there are two kinds of parity drives, where P drive means slope 0 and Q for slope 1. Whenever any single data disk fails, the conventional way to reconstruct a failed disk is merely using P drive while the optimal way is using equal number of parities from P and Q drives, maximizing the overlapping data, as shown in Figure 2 [17].

(ii) RDP+ Construction. As proposed in RAID-6Plus, the original P and Q drives of RDP code are kept to maintain base reliability of double-failure tolerance. Much more attention is paid to the coding of the third redundant drive X. Getting insights from the optimal reconstruction of single failure in RDP, we understand the hybrid use of equal numbers of P parities and Q parities can maximize the overlapping data for single failure rebuild. Thus an attempt has been made to employ short combination with some two elements involved

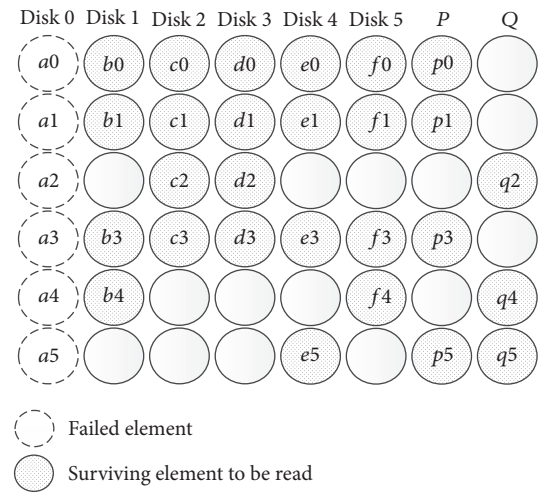


Figure 2: The optimal reconstruction sequence of single failure in RDP.

in optimal single rebuild to encode for X parity, as shown in Figure 3.

In the view of X drive in RDP+, nearly all the data drives are covered, because any new parity in the third parity drive X is the XOR of some two data elements. Those data pairs or tuples to construct X parity are specially chosen to satisfy fast reconstruction. For example, $a0$ and $a1$ of disk #0 are, respectively, included in $x0$ and $x1$, which will be used in the reconstruction of disk #0. In other words, disk coverage in some way guarantees the even and balanced distribution of speedup effect on multidisks. Those short combinations in X drive constructed by data pairs are specially chosen to satisfy fast reconstruction. Similar algorithms to those in [18–20] can be easily constructed to find proper short combinations for X drive.

(iii) Single Failure Rebuild in RDP+. Regarding single failure reconstruction, for example, if *disk#0* fails, reconstruction with the participation of related short combinations in X will occur.

As shown in Figure 4, we use $x0$ and $f0$ to recover $a0$, $x1$, and $e1$ for $a1$. With the help of $p3$ and $p4$, respectively, $a3$ and $a4$ are reconstructed in the direction of slope 0 while $a2$ and $a5$ are, respectively, recovered from slope −1

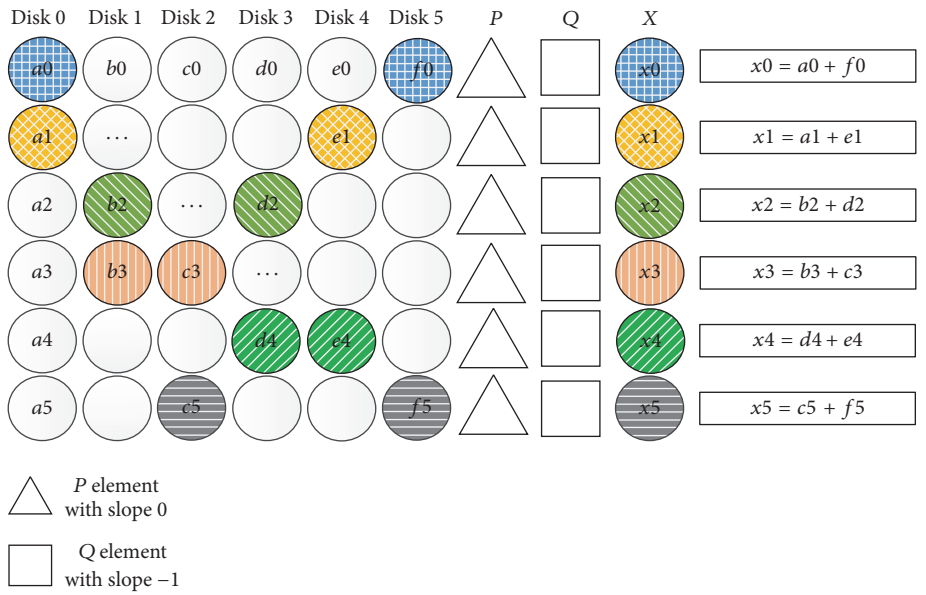


Figure 3: Encoding for X drive in RDP+.

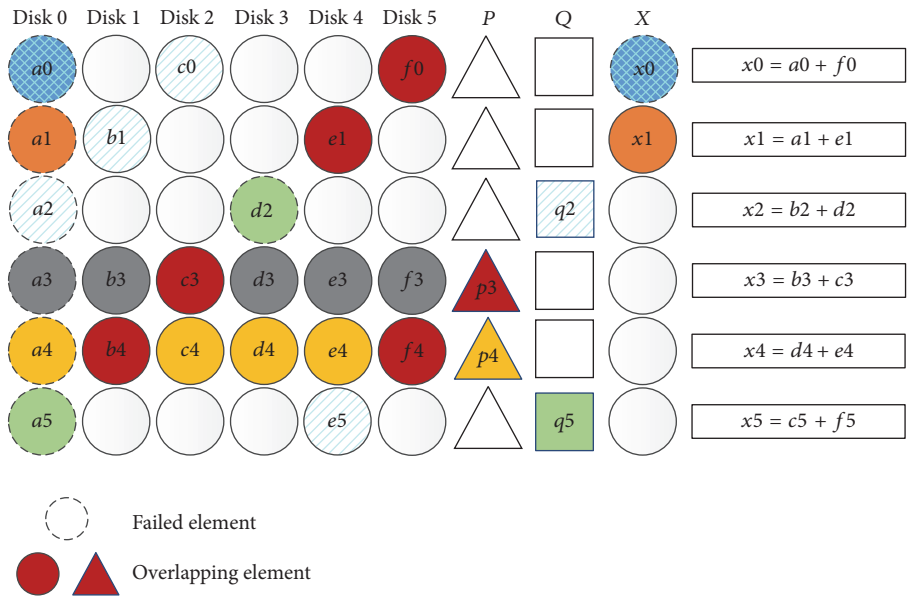


Figure 4: Rebuilding single failure with X elements.

with $q2$ and $q5$. In total, there are 22 elements needed to be read, comparing with 27 element reads of RDP optimal recovery.

In the views of double failures, RDP+ maintains system reliability in two ways. First and foremost, RDP is included in RDP+; therefore there will be no data loss whenever any double failures happen.

Additionally, with shorter reconstruction window, some double-failure situations previous in traditional RAID-6 system could be converted to independent single failures, thus eliminating vulnerability undergone by the system. Also, the same way of using short combinations in X can be applied to save data reads for double-failure scenarios.

In fact, triple failures happen at an extremely tiny probability; therefore the third parity drive in traditional codes only exists for extreme cases. According to Reliability Equation in [26], RDP+ could convert a portion of triple-failure situations in traditional coding schemes to lower possibility, leaving unconvertible triple failures at a negligible level.

3.3. X-Code+ over X-Code. Unlike aforementioned RAID-6 codes, which are all horizontally aligned, X-code [6] stands out as a vertical code and has the unique property in update complexity, which is denoted by the penalized writes to parity caused by a write request to a single data element. Additionally, nonvolatile memory (NVM) is gaining

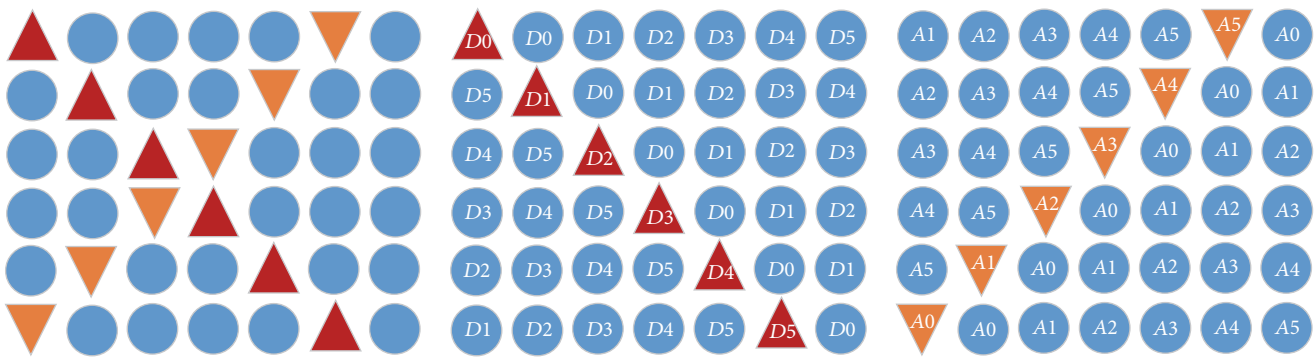


FIGURE 5: (a) presents the layout of modified X-code. (b) and (c), respectively, give the detailed construction of D parity and A parity.

popularity and much sensitive to write operation [12]. With its born optimality on update penalty, X-code is very suitable to be used in nonvolatile memory systems for its optimal update complexity to mitigate write operations. However, no extension based on X-code exists, leaving us possibility of extending X-code with the proposed RAID-6Plus methodology.

Nevertheless, since its different data layout, what has been done to RDP cannot simply be applied to RDP to get the extended code. A particular modification of data layout on X-code is needed.

(i) Layout Modification of Original X-Code. Originally, all parity elements in X-codes are aligned downward horizontally. When being extended, the layout of X-code needs to be modified to maintain storage efficiency. Therefore, all parity elements in *the shape of* X among the data elements are aligned and one more column of data elements is added for balance, as shown in Figure 5.

In this way, original X-code of size $p \times p$ has been modified to size $(p-1)p$ and with all the following properties preserved: (1) parity construction, (2) update complexity, and (3) MDS property. The things changed are array size and element layout.

The objective for doing so is to achieve the following:

(1) *Base reliability*: the modified X-code is aimed at maintaining a lower bound of reliability, denoted as "base reliability," given that RAID-6 is widely deployed in various storage system for double-failure concern.

(2) Optimal update complexity before extension so as to mitigate media wear-out penalized with write operations.

(ii) X-Code+ Construction. The modified X-code is extended by adding one more drive of parity as existing extensions do, but in a new and more practical way.

Note that different from RDP+, load balance is taken into consideration when constructing X-code+. It is determined by the layout of X-code, where data elements and parity elements are stored together in each drive. Therefore X-code+ is intended to achieve a reasonable compromise among (1) reliability, (2) performance, and (3) load balance, where,

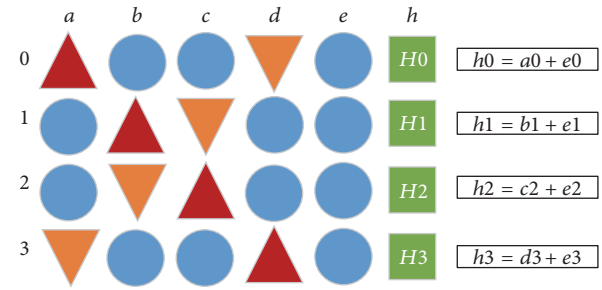


FIGURE 6: The coding for parity elements in H drive.

on the one hand, we maintain a basic reliability guarantee to accommodate nonnegligible double-failure, while, on the other hand, reconstruction window is shortened to degrade failure and offload bottleneck access and reduce user wait.

In the proposed X-code+, the added parity drive is denoted as H *drive* for clarity, whose purpose is to reduce reconstruction window and balance load for single failure and whose parity is constructed *horizontally*. Though H *drive* is a pure parity drive, free of user access; it plays a key role in various failure modes.

In terms of constructing parity elements in H drive, two-element tuples are selected horizontally. By getting insight from the optimal single failure recovery of modified X-code, it is aimed at (1) minimizing total data reads by reusing overlapping data and (2) balancing load by offloading bottleneck workload to the H drive as much as possible.

With the concrete example in Figure 6, the proposed construction of parity elements in H drive is illustrated.

As can be seen from Figure 6, parity elements in H drive are aligned vertically and each one is the XOR sum of a two-element tuple; for example, $h0$ is the XOR sum of $a0$ and $e0$. In order to cover all existing drives when any of them fails, some elements are included in the components of H drives. For example, $a0$ in h0 is included, which will be used in face of column a. Likewise, other drives are covered by introducing some elements. To pay attention, E drive is the only one that stores only data to bear much user access; therefore we have all its elements ($e0$, $e1$, $e2$, and $e3$) as components for H parity.

Table 2: Recover $a0$ with parity of different length in Figure 8.

Parity	Parity type	Parity length	Recovery sequence for $a0$	Data needed
$H0 = a0 + e0$	Horizontal	2	$h0$, $e0$	2
$A0 = b0 + d2 + e3$	Diagonal	3	$b0$, $d2$, $e3$	3

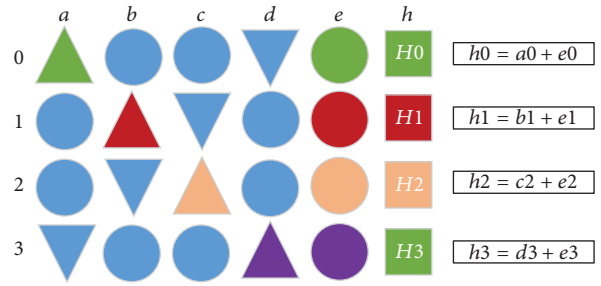


Figure 7: The general construction of H parity as the XOR sum of D parity with its corresponding element in E drive.

In other words, storage drive coverage in this way guarantees as much as we can the even and balanced distribution of speedup effect on multiple drives.

In essence, H parity elements are combinations with the length shorter than original D (diagonal) parity or A (antidiagonal) parity so that its components could be recovered with fewer data. For example, any D or A parity consisted of 3 elements while H parity is comprised of 2 elements. Furthermore, the gap will be bigger and pronounced with the array size growing up.

In terms of which elements should be selected to be components, a simple and straightforward way is introduced by adding one element from D parity and its corresponding element horizontally in E drive, as shown in Figure 7.

Because D parity and A parity are symmetrical, therefore it is acceptable to use either D parity or A parity.

(iii) Single Failure Rebuilt in X-Code+. X-code+ is intended to strike a balance between reliability and performance and load balance. This property is better illustrated in the face of single failure reconstruction. It has been reported that 99.75% of recoveries are for the single disk failure [24]. Accordingly, the performance of recovery for single disk failure is of high importance to reduce window of vulnerability.

Regarding single failure reconstruction, for example, if column a fails, reconstruction with the participation of related parity in H drive will occur. As shown in Figure 8(a), before extension, $a0$ and $a1$ are recovered along the diagonal direction while $a2$ and $a3$ are fixed with the help of their corresponding antidiagonal elements.

In contrast as shown in Figure 8(b), $h0$ and $f0$ are used to recover $a0$. With the help of parity $D2$ and $D3$, respectively, $a2$ and $a3$ are reconstructed in the direction of slope 1 while parity $A0$ is, respectively, recovered from slope −1 with element $b3$, $d1$, and $e0$. Therefore, with the help of parity $h0$, $f0$ is recovered horizontally instead of diagonally and meanwhile, the overlapped $e0$ is preserved.

In order to better measure reconstruction performance, we present a comparison on total data reads and bottleneck drive.

(iv) Total Data Reads. In the erasure coding field, reducing total amount of data reads in the recovery is the main consideration behind code design. There are examples in non-MDS (Maximum Distance Separable) [5] codes such as Pyramid code [28]. Further, many excellent works of optimization are also measured by total data reads, like Khan et al. [29]. In this way, it could be seen that X-code+ is able to reduce total data reads for single failure recovery from 10 to 8.

The reason behind total data reads reduction is because of the following:

(1) *H parity* with shorter length is used to replace original *D parity* or *A parity* with longer length; thus less elements are needed, as shown in Table 2.

(2) Overlapped elements are preserved and reused with the use of H parity. This is our consideration in the construction of parity elements in H drive. For example, in Figure 3, $e0$ is one of the overlapped elements and in order to preserve and reuse it after extension, we have $e0$ in $h0$ in Figure 8.

In this way, X-code+ is conducive to saving total reads. This effect could be greater when array size grows bigger and bigger.

(v) Load Balance. There are two flaws to use total data reads as an indicator for recovery performance: (1) total data read is a static metric with particular codes. Their recovery performance is determined as long as the codes are designed. (2) Given the distribution and parallel IO across multidrives, minimizing the total amount of data accessed for the recovery does not necessarily translate into minimal recovery time [23, 30].

Therefore, with multiple drives serving the read requests for a recovery with parallel I/O, it is the service time of the disk that reads the largest amount of data that determine the recovery time.

In terms of this, it can be seen in Figure 8(a) that the bottleneck before extension is *column d* with 4 elements read while the bottleneck after extension is reduced to 2 elements read in *columns c, d*, or *e*, as shown in Figure 8(b). Therefore the proposed X-code+ is able to offload recovery IO with help of newly added *H parity*. This would translate directly into recovery performance.

In summary, X-code+ outperforms modified X-code on both total data reads and load balance.

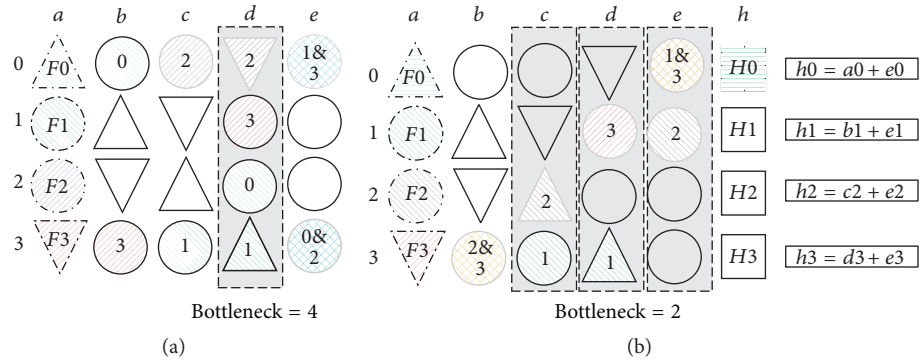


Figure 8: (a) Recovery of column a before extension; (b) recovery of column a after extension.

4. Evaluation

In this section, RDP+ and X-code+ along with its counterpart codes are compared at the same cost in the aspects of parity computation complexity, update penalty, reconstruction cost, and storage ratio. The basic size for STAR and modified X-code is $(p-1)p$ and $(p-1)(p-1)$ for RTP. Further, a Q metric is proposed to denote the induced performance improved per cost to validate the proposed RAID-6Plus methodology.

4.1. Encoding Complexity. The encoding process is complicated and influenced by many aspects. However in theory, parity computation complexity is reasonably used to denote the encoding complexity. Hereby, given XOR as a basic operation, XORs per element is used to quantitatively represent parity computation complexity among different codes.

The total XORs are the sum of XORs for different parity types. For example, the number of XORs for writing a stripe in RDP+ consisted of XORs for row parity, diagonal parity, and X parity, while those of X-code+ are of XORs for diagonal parity, antidiagonal parity, and horizontal parity, given H parity could be calculated with existing D parity cached and data. Thus, we have

$$\begin{aligned} A_{\text{RDP+}} &= \frac{\text{XOR}_{\text{row}} + \text{XOR}_{\text{dia}} + \text{XOR}_{x}}{\text{Blocknum}} \\ &= \frac{(p-1)(p-2) + (p-1)(p-2) + p - 1}{(p-1)(p-1)} \\ &= \frac{2p-3}{p-1} = 2 - \frac{1}{p-1}, \\ A_{X\text{-code+}} &= \frac{\text{XOR}_{\text{Dia}} + \text{XOR}_{\text{anti-dia}} + \text{XOR}_{\text{Hori}}}{\text{Elementnum}} \\ &= \frac{(p-1)(p-3) + (p-1)(p-3) + (p-1)}{(p-1)(p-2)} \\ &= 3 - \frac{1}{p-2}. \end{aligned} \tag{1}$$

Similarly, we can get that of RTP is $A_{\text{RTP}} = 3 - 3/(p-1)$ and $A_{\text{STAR}} = 3 - 1/(p-1)$ for STAR. The results are shown in Figure 9.

Obviously, RDP+ uses least XORs per data block for parity computation and is fastest. The difference is more obvious when disk array size is smaller, which is the typical case of most storage systems. During encoding process, the complexity of X-code+ is between RTP and STAR code. The differences in between will become less and less with array size growing bigger. In the best case, X-code+ is 3.1% faster than STAR.

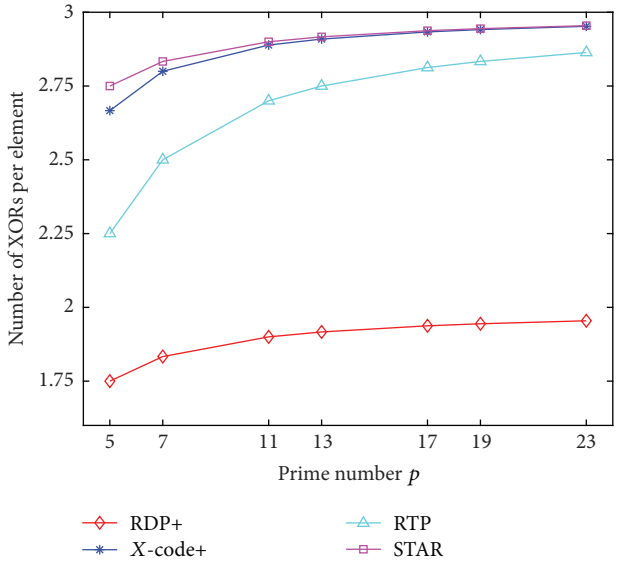


Figure 9: Parity computation complexity.

4.2. Update Penalty. Redundancy incurs update penalty for consistency between data and its related parity. Hereby, the update penalty is measured by number of introduced writes except writing on data itself. For example, in original X-code, any update on a single data block will result in two more writes to, respectively, its corresponding diagonal parity and antidiagonal parity; thus its update penalty is 2. Through similar calculation [31], the update penalty of RTP turns out to be $U_{\text{RTP}} = 3 - 2/(p-1) + 2/(p-1)^2$ [8] while STAR is $U_{\text{STAR}} = 5 - 4/p$ in [32].

For RDP+, update penalty of all data elements requires

$$\begin{aligned} &[(p-1)(p-1) - 2(p-1) - (p-2)] \times 2 + 2 \\ &\quad \times (p-1) \times 3 + (p-2) \times 1 = 2p^2 - 3p + 2. \end{aligned} \tag{2}$$

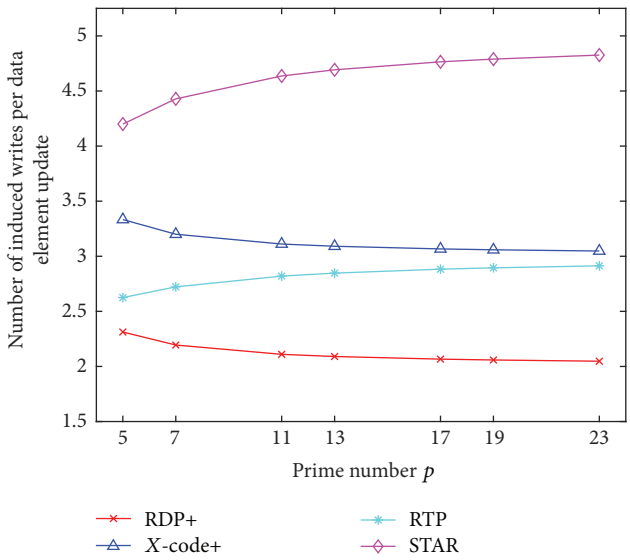


Figure 10: Update penalty.

Thus, the update penalty per block is

$$U_{\text{RDP+}} = \frac{\left(2p^2 - 3p + 2\right)}{(p-1)^2} = 2 + \frac{1}{p-1} + \frac{1}{(p-1)^2} \tag{3}$$

which is smaller than RTP and STAR. Likewise, update penalty of all data elements for X-code+ is

$$\begin{aligned} U_{X\text{-code+}} &= \frac{3\left[(p-1)^2 - 2(p-1)\right] + 4(p-1)}{(p-1)\times(p+1) - 3(p-1)} \\ &= 3 + \frac{1}{p-2}. \end{aligned} \tag{4}$$

Comparison results are plotted in Figure 10. Significantly, RDP+ bears least update penalty, and it converges toward optimal update penalty of double-failure-tolerant codes with bigger array size while X-code+ converges toward optimal update penalty of triple-failure-tolerant codes. For example, when p equals 11, RDP+ has 2.11 update penalties compared with 2.82 of RTP and 4.636 of STAR. The difference will be more obvious when disk array size grows bigger. The reason is the proposed RDP+ methodology that has least parity nesting in the third parity drive, while RTP and STAR have more parity nesting and use full combinations with larger length, therefore incurring more update penalty.

4.3. Reconstruction Cost for Single Failure. Specifically, the reconstruction performance for single failure is evaluated on both total data read and load balance.

In terms of single failure, because RTP and RDP share exactly the same reconstruction sequences, they have the same reconstruction cost, which are, respectively $(p-1)(p-1)$, with conventional recovery and $(3/4)(p-1)^2$ with optimal recovery. Similarly, STAR rebuilds a failed drive with the same cost of EVENODD, which is $(p-1)p$ conventionally and $(3p+1)(p-1)/4$ by optimal recovery.

When a drive fails in RDP+, two elements are recovered by two X parities and half of the rest elements are, respectively, rebuilt with P and Q parity as shown in Figure 5. Therefore, we compute data elements read for rebuild in RDP+ as

$$\begin{aligned} &2\times 2 + \frac{(p-1)-2}{2}\times(p-1) + \frac{(p-1)-2}{2}\times(p-1) \\ &\quad - \frac{(p-1)-2}{2}\times\frac{(p-1)-2}{2} - 2 \\ &= 2 + \frac{(p-3)(3p-1)}{4}. \end{aligned} \tag{5}$$

Initially, the total data read of modified X-code is

$$\begin{aligned} T_{M\text{-}X\text{-code}} &= (p-1)(p-2) - \frac{p-1}{2}\times\frac{p-3}{2} \\ &= \frac{p-1}{4}(3p-5). \end{aligned} \tag{6}$$

With the help of *H parity* participating in reconstruction, total data reads could be reduced with shorter parity length and reuse of overlapping elements. Therefore, through calculation, the total data read of X-code+ is achieved as

$$\begin{aligned} T_{X\text{-code+}} &= 2 + (p-2)(p-2) - 1 - \frac{p-1}{2}\times\frac{p-3}{2} \\ &= \frac{3(p-1)(p-3)}{4} + 2. \end{aligned} \tag{7}$$

Hereby results normalized by the total data read of modified X-code are shown in Figure 11.

Clearly, X-code+ reads the least data and RDP+ is the second least. For example, when p is 7, RDP+, respectively, have a normalized speedup of 33.4%, 11.9%, 47.7%, and 26.2%, respectively, over RTP under the conventional rebuild (labeled as RTP_C), RTP under the optimal rebuild (labeled as RTP_O), STAR under that conventional rebuild (labeled at STAR_C), and STAR under optimal rebuild (labeled at STAR_O), respectively.

The secret of the proposed methodology for single erasure rebuild is that it uses short combinations in the third redundant drive to minimize elements needed and meanwhile reuse overlapping elements as much as possible by optimal reconstruction sequence of RDP.

(i) Load Balance for X-Code+. Upon normal user access, RTP and STAR will experience the hot-drive effect because parity elements are placed in drives independent of data, causing serious update bottleneck. In contrast, X-code+ employs a hybrid placement policy by mixing diagonal and antidiagonal parity with data and meanwhile storing horizontal parity independently, thus mitigating the hot-drive effect and balance access.

When failure happens, X-code+ reconstructs failed drives with the help of horizontal parity to alleviate the recovery bottleneck with most IO on it by offloading some load to the *H drive*, as shown in Figure 8. Under some

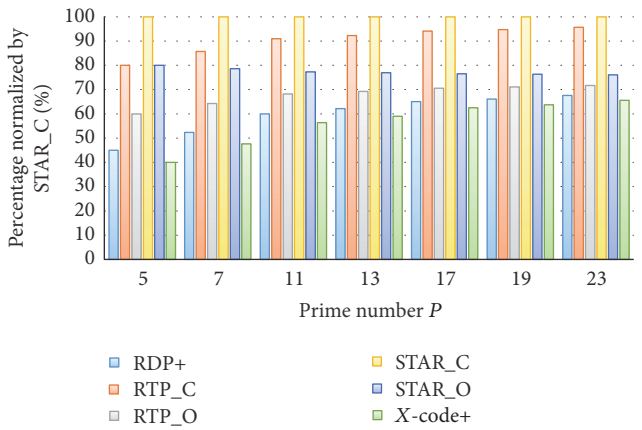


Figure 11: Data reads for rebuilding single failure.

specific reconstruction sequence, the bottleneck load could be reduced from $p-1$ to $p-3$.

In this way, in both normal and failure situations, X-code+ provides better load balance among its peer codes.

4.4. Storage Overhead and Reliability. RTP and STAR are both MDS codes [5] while RDP+ and X-code+ are non-MDS. Regarding storage overheads, all the three codes have three drives dedicated to parities. In terms of absolute failures, RTP and STAR are strictly triple-failure tolerant while both RDP+ and X-code+ can tolerate only double failures at device level. But they provide flexible and higher relative reliability by shorter single failure reconstruction. This effect is hard to be explicitly characterized because there are not any plausible reliability model for RAID systems.

4.5. Q Metric for Reconstruction. Better performance comes at some cost. Usually, the gains and the pains are independently evaluated, without any indicator to suggest their correlation. However, the Q metric we propose intend to bridge this gap and is truly an indicator of improvement correlated with cost we spent. Hereby we use RDP+ as an example to illustrate.

Essentially, coding schemes like RTP, STAR, and the proposed RDP+ are extending existing RAID-6 array codes by adding extra parity drive. In order to evaluate the validity and utility of adding extra parity drive, a Q metric is proposed to denote the induced benefit per cost. The higher the Q value is, the more competitive and useful the method will be. Exactly in this paper,

$$Q = \frac{\text{Induced benefit}}{\text{Cost}} = \frac{\text{Speedup of single failure rebuild}}{\text{ratio of extra parity over data drives}}. \quad (8)$$

When adding a third parity drive, we have three ways to encode the third parity drive, as mentioned in Table 1. RTP is using full stripe while the proposed RAID-6Plus is short striped. We will compare RTP, RAID-6Plus, mirroring for single drive, and mirroring for all drives with Q metric and take RDP_optimal as the baseline performance.

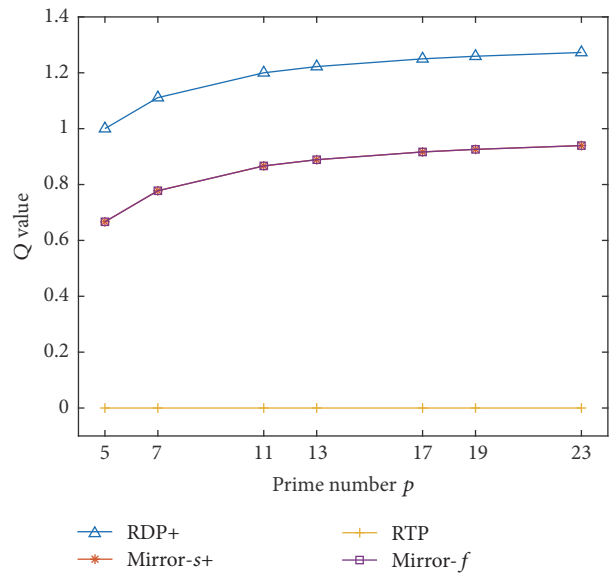


Figure 12: Q comparison of four ways of adding third parity drive.

$$Q_{\text{RTP}} = \frac{\left[(3/4)(p-1)^2 - (3/4)(p-1)^2\right] \div (3/4)(p-1)^2}{1/(p-1)} = 0,$$

$$Q_{\text{RAIS-}P} = \frac{\left[(3/4)(p-1)^2 - (2+(p-3)(3p-1)/4)\right] \div (3/4)(p-1)^2}{1/(p-1)} = \frac{4(p-2)}{3(p-1)},$$

$$Q_{\text{Mirror\_}s} = \frac{\left[(3/4)(p-1)^2 - 3(p-1)(p-2)/4 - 1\right] \div (3/4)(p-1)^2}{1/(p-1)} = \frac{3p-7}{3p-3}, \quad (9)$$

$$Q_{\text{Mirror\_}f} = \frac{\left[(3/4)(p-1)^2 - (p-1)\right] \div (3/4)(p-1)^2}{1} = \frac{3p-7}{3p-3}.$$

The results are shown in Figure 12.

In this way, it can be clearly seen that the existing RTP has no contribution in accelerating single failure rebuild with a zero Q, which is the very motivation of RAID-6Plus. Among possible way to encode the third parity drive, RAID-6Plus prevails. The reason behind this is RAID-6Plus has nearly an even and balanced coverage for all data drives and short combinations in X drive use as less data elements as possible.

In short, Q metric is proving that, unlike RTP and STAR, RAID-6Plus is a valid way to extend RDP from the perspective of single failure reconstruction.

5. Related Work

Typically, in erasure codes, there are two basic arithmetic to be in use: XOR and Galois Field. XOR-based codes are faster than GF-based codes while GF-based code has wider flexibility in code construction [5]. RAID-6Plus, as an extension on the basis of XOR-based codes, is still XOR-based, such as RTP [8] and STAR [27]. RAID-6Plus is faster with less complexity incurred in parity computation.

MDS codes like RS code are traditional focus of storage reliability for they offer further reliability boundary, that is, absolute fault tolerance [5]. Both RTP and STAR are MDS codes, but they are unable to accelerate single failure reconstruction. Non-MDS codes are thus invented. A typical example is the LRC codes from Microsoft [33]. However, LRC is horizontal codes and GF-based while RAID-6Plus is array code and XOR-based.

Efforts have also been made to accelerating single failure. They focus either on finding optimal reconstruction sequences [17–20] or on load-balancing [23, 30]. However all these work are limited to the given codes, without modification or extension. To our knowledge, RAID-6Plus is first to extend given codes from the perspective of speedup single failure instead of providing reliability boundary. Further, RAID-6Plus [14] is orthogonal with previous work above and can be integrated together to further shorten reconstruction window.

6. Conclusions

The existing methodology of extending RAID-6 codes considers only the fault tolerance level by utilizing the third redundant drive to recover from triple-failure scenarios. As a result, no contribution is made to speed up rebuilding single failures, and consequently, the third parity drive is almost idle.

This paper proposes RAID-6Plus, allowing the reuse of overlapped elements during reconstruction to balance the reliability and performance of the resulting coding scheme. By applying the proposed methodology to RDP and X-code, respectively, we generate two new coding schemes named RDP+ and X-code+ to provide a better balance between performance and reliability. The performance evaluation indicated that RAID-6Plus exhibited (1) a better system performance with no extra cost compared to RTP and STAR and (2) an enhanced reliability compared to RAID-6. With the proposed Q-metric, a detailed justification of the proposed methodology is provided.

In short, RAID-6Plus held the potential of practical uses in modern disk systems, flash-based systems, and even hybrid storage systems on any array codes. As for the future work, it would be interesting to investigate how RAID-6Plus should be applied in real storage systems and also how such algorithm could be optimized in terms of flash memory. For example, X-code+ would be helpful to mitigate the write amplification problem of flash memory with less update penalty in providing system reliability.

Disclosure

This work is an extended version of the conference papers presented at 4th International Conference on Computer Science and Network Technology (ICCSNT) [21] and 9th Asia-Pacific Services Computing Conference, APSCC 2015.

Competing Interests

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

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant nos. 61433019, U1435217, 61232003, 61502514, 61402503, and 61402501 and National High Technology Research and Development 863 Program of China under Grant 2015AA015305.

References

[1] A. Ma, R. Traylor, F. Douglis et al., "RAIDShield: characterizing, monitoring, and proactively protecting against disk failures," *ACM Transactions on Storage (TOS)*, vol. 11, no. 4, article 17, 2015.

[2] J. S. Plank, "A tutorial on Reed-Solomon coding for fault-tolerance in RAID-like systems," *Software, Practice & Experience (SPE)*, vol. 27, no. 9, pp. 995–1012, 1997.

[3] M. Blaum, J. Brady, J. Bruck, and J. Menon, "EVENODD: an efficient scheme for tolerating double disk failures in RAID architectures," *IEEE Transactions on Computers*, vol. 44, no. 2, pp. 192–202, 1995.

[4] C. Lueth, "RAID-DP: network appliance implementation of RAID double parity for data protection," Tech. Rep. 3298, Network Appliance, 2004.

[5] J. S. Plank, "T1: erasure codes for storage applications," in *Proceedings of the 4th USENIX Conference on File and Storage Technologies (FAST '05)*, San Francisco, Calif, USA, December 2005.

[6] M. Blaum, T. Cortes, and H. Jin, "The EVENODD code and its generalization," *High Performance Mass Storage and Parallel I/O*, pp. 187–208, 2001.

[7] C. Huang and L. Xu, "STAR: an efficient coding scheme for correcting triple storage node failures," *IEEE Transactions on Computers*, vol. 57, no. 7, pp. 889–901, 2008.

[8] A. Goel and P. Corbett, "RAID triple parity," *ACM SIGOPS Operating Systems Review*, vol. 46, no. 3, pp. 41–49, 2012.

[9] Y. Wang, G. Li, and X. Zhong, "Triple-star: a coding scheme with optimal encoding complexity for tolerating triple disk failures in raid," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 3 A, pp. 1731–1742, 2012.

[10] Y. Wang and G. Li, "Rotary-code: efficient MDS array codes for RAID-6 disk arrays," *WSEAS Transactions on Computers*, vol. 8, no. 12, pp. 1917–1926, 2009.

[11] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson, "RAID: high-performance, reliable secondary storage," *ACM Computing Surveys (CSUR)*, vol. 26, no. 2, pp. 145–185, 1994.

[12] A. Amer, D. D. E. Long, and S. J. Thomas Schwarz, "Reliability challenges for storing exabytes," in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC '14)*, pp. 907–913, IEEE, Honolulu, Hawaii, USA, February 2014.

[13] B. Schroeder and G. A. Gibson, "Disk failures in the real world: what does an MTTF of 1, 000, 000 hours mean to you?" *FAST*, vol. 7, pp. 1–16, 2007.

[14] M.-Z. Deng, Y. Ou, N. Xiao et al., "RAID-6Plus: a fast and reliable coding scheme aided by multi-failure degradation," in *Advances in Services Computing*, vol. 9464, pp. 210–221, Springer, Berlin, Germany, 2015.

[15] J. S. Plank and M. Blaum, "Sector-disk (SD) erasure codes for mixed failure modes in RAID systems," *ACM Transactions on Storage*, vol. 10, article 4, 2014.

[16] A. Leventhal, "Triple-parity RAID and beyond," *Queue*, vol. 7, no. 11, 2009.

[17] L. Xiang, Y. Xu, J. C. Lui, and Q. Chang, "Optimal recovery of single disk failure in RDP code storage systems," in *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '10)*, pp. 119–130, ACM, New York, NY, USA, 2010.

[18] L. Xiang, Y. Xu, J. C. S. Lui, Q. Chang, Y. Pan, and R. Li, "A hybrid approach to failed disk recovery using RAID-6 codes: algorithms and performance evaluation," *ACM Transactions on Storage*, vol. 7, article 11, 2011.

[19] Y. Zhu, P. P. C. Lee, L. Xiang, Y. Xu, and L. Gao, "A cost-based heterogeneous recovery scheme for distributed storage systems with RAID-6 codes," in *Proceedings of the 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN '12)*, Boston, Mass, USA, June 2012.

[20] O. Khan, R. Burns, J. Plank, and W. Pierce, "Rethinking erasure codes for cloud file systems: minimizing I/O for recovery and degraded reads," in *Proceedings of the 10th USENIX Conference on File and Storage Technologies (FAST '12)*, San Jose, Calif, USA, February 2012.

[21] M. Deng, L. Zhu, N. Xiao, Z. Chen, and F. Liu, "X-code+: a compromised coding scheme with smaller rebuild window and load-balance," in *Proceedings of the 4th International Conference on Computer Science and Network Technology (ICCSNT '15)*, December 2015.

[22] J. S. Plank, M. Blaum, and J. L. Hafner, "SD codes: erasure codes designed for how storage systems really fail," in *Proceedings of the 11th USENIX Conference on File and Storage Technologies (FAST '13)*, San Jose, Calif, USA, February 2013.

[23] X. Luo and J. Shu, "Load-Balanced recovery schemes for single-disk failure in storage systems with any erasure code," in *Proceedings of the 42nd Annual International Conference on Parallel Processing (ICPP '13)*, pp. 552–561, Lyon, France, October 2013.

[24] E. Pinheiro, W.-D. Weber, and L. A. Barroso, "Failure trends in a large disk drive population," *FAST*, 2007.

[25] R. Bez, E. Camerlenghi, A. Modelli, and A. Visconti, "Introduction to flash memory," *Proceedings of the IEEE*, vol. 91, no. 4, pp. 489–502, 2003.

[26] J. G. Elerath and J. Schindler, "Beyond MTTDL: a closed-form RAID 6 reliability equation," *ACM Transactions on Storage*, vol. 10, no. 2, article 7, Article ID 2577386, 2014.

[27] C. Huang and L. Xu, "STAR: an efficient coding scheme for correcting triple storage node failures," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 57, no. 7, pp. 889–901, 2008.

[28] C. Huang, M. Chen, and J. Li, "Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems," *ACM Transactions on Storage*, vol. 9, no. 1, article 3, 2013.

[29] O. Khan, R. Burns, J. Plank, W. Pierce, and C. Huang, "Rethinking erasure codes for cloud file systems: minimizing I/O for recovery and degraded reads," in *Proceedings of the 10th USENIX Conference on File and Storage Technologies (FAST '12)*, San Jose, Calif, USA, 2012.

[30] Y. Fu, J. Shu, and X. Luo, "A stack-based single disk failure recovery scheme for erasure coded storage systems," in *Proceedings of the 33rd IEEE International Symposium on Reliable Distributed Systems (SRDS '14)*, pp. 136–145, IEEE, Nara, Japan, October 2014.

[31] R. Hu, G. Liu, and J. Jiang, "An efficient coding scheme for tolerating double disk failures," in *Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications (HPCC '10)*, pp. 707–712, September 2010.

[32] H. Rongdong, L. Guangming, and J. Jingfei, "An efficient coding scheme for tolerating double disk failures," in *Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications (HPCC '10)*, Melbourne, Australia, 2010.

[33] C. Huang, H. Simitci, Y. Xu et al., "Erasure coding in windows azure storage," in *Proceedings of the USENIX Annual Technical Conference (USENIX ATC '12)*, Boston, Mass, USA, June 2012.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 7871502, 9 pages
https://doi.org/10.1155/2017/7871502

Research Article

Recommending Locations Based on Users' Periodic Behaviors

Bing Xu, Zhijun Ding, and Hongzhong Chen

Department of Computer Science and Technology, Tongji University, Shanghai, China

Correspondence should be addressed to Zhijun Ding; dingzj@tongji.edu.cn

Received 23 September 2016; Revised 11 December 2016; Accepted 22 December 2016; Published 21 February 2017

Academic Editor: Qingchen Zhang

The research of location recommendation system is an important topic in the field of LBSN (Location-Based Social Network). Recently, more and more researchers began focusing on researching how to recommend locations based on user's life behavior. In this paper, we proposed a new model recommending locations based on user's periodic behaviors. In view of multiple periodic behaviors existing in time series, an algorithm which can mine all periods in time series is proposed in this paper. Based on the periodic behaviors, we recommend locations using item-based collaborative filtering algorithm. In this paper, we will also introduce our recommendation system which can collect users' GPS trajectory, mine user's multiple periods, and recommend locations based user's periodic behavior.

1. Introduction

LBSN (Location-Based Social Network) is a new kind of social network which combines the user's friendship and the user's position. There are two hot research hotspots in the filed of LBSN: recommendation system and mining user's life pattern. For recommending system, the user can get more suggestion for deciding where he can choose to go to. For mining life pattern, the recommendation system can better understand the use's preference. Periodic behavior is a kind of life pattern. In this paper, we proposed a new kind of recommendation model based on the user's periodic behaviors. For solving the problem of mining multiple periods, we proposed a new algorithm which can mine multiple periods in the time sequence.

The research of recommendation systems in the field of LBSN has attracted many researchers' attention and produced a lot of research results. Currently, the recommendation system of LBSN is divided into good friends recommended [1], location recommended [2], activities recommended [3], and event recommended [4]. In the field of friend recommendation, Papadimitriou et al. proposed a friend recommendation algorithm based on user social network [1]. For location recommendation, Bellotti et al. proposed a personalized location recommendation algorithm based on social network and location influence [3]. The activities recommendation is recommending a place for users' preferred activities. Bellotti et al. proposed a system which can recommend the preferred location of activities. The event recommendation is a special case of activities recommending; the main research directions in the field is event detection. Papadimitriou et al. proposed a [2] position information label incident detection algorithm. Furthermore, Gao et al. [5] proposed a location recommendation model which can solve the cold start problem. Zhang et al. [6] proposed a personalized and efficient geographical location recommendation framework called iGeoRec. Huang [7] proposed a novel context similarity measure to quantify the similarity between any two contexts and develop three context-aware collaborative filtering methods for recommending locations. Yuan and Li [8] proposed a location recommendation algorithm based on temporal and geographical similarity in location-based social networks.

With the gradual accumulation of user' location data, the research of users' life behavior has aroused the interest of some researchers and has produced the relevant research results. Rekimoto et al. proposed a user interest point travel sequence mining algorithm based on GPS trajectory and proposed an algorithm which can predict the next destination based on the user's current location [9]. Ye et al. proposed

a user behavior pattern mining algorithm based on WIFI's location monitoring technology [10]. Lahiri and Berger-Wolf proposed a LP-Mine algorithm which can mine user's behavior patterns from the original GPS trajectory data [11].

Periodic behavior is a kind of life behavior. The user arriving in a certain area periodically is called a periodic behavior. For example, a user goes shopping every weekend and a user reaches the company for working every Friday to Monday. In order to obtain periods, we should use some period mining algorithm. In this paper, we proposed a new algorithm for mining periods and proposed a new model for recommending locations based on users' periodic behaviors.

Recently, some researchers have proposed some period mining algorithms. Li et al. used the periodogram and self-correlation method for mining the user periodic behaviors [12, 13], and to combine the periodic behavior, the authors proposed a kind of location predicting model based on the thought of probability. Wang proposed the basic thought of periodogram [14]. After several years of development, this algorithm has been successfully applied in different areas for mining periods, for example, hydrological time series analysis [15] and medical time series analysis [16]. The basic idea of the period diagram is based on the power density spectrum estimation which is based on Fourier transform. Because of the leakage of the spectrum, this algorithm is not accurate for mining all periods. In order to avoid this problem [17], windowing technique is proposed by the researcher, but this technology is not accurate enough. Self-correlation method has been proposed. Compared with the periodogram, this algorithm can solve the problem of mining periods from arbitrary length time sequence. In addition, Parthasarathy et al. proposed an algorithm for obtaining [18] time series based on cross entropy; the algorithm can get into one of the most significant periods of time series. By using this algorithm, it is possible to find out the periodic behavior of the users in the purchase of electronic goods. Wang et al. also proposed a time series period mining algorithm [19], which is designed based on the frequent features of periodic behavior and the basic idea of the structure. Xu et al. proposed an [20] algorithm model of hydrologic sequence period based on Mexhat wavelet; this model can be used to dig out periodic behavior from the monthly rainfall in Shanghai.

In real life, some users go to a certain area with multiple periods. For example, a user goes to an area not only every Friday but also every two weeks on Monday. In this paper, we proposed a period acquisition algorithm which can mine multiple periods in time sequence correctly. In our research, we found that there are multiple periods existing in some time sequence generated by users. For example, a teacher not only comes to a campus every Monday but also comes to this campus on Tuesday every two weeks. For this teacher, there are two periodic behavior in his daily work behavior. The classical algorithm, periodogram and self-correlation function, cannot mine all correct periods because multiple periods affect each other while mining periods. In our experience, they cannot mine multiple periods in the time sequence generated by restrict multiple periodic behaviors. The main thought of periodogram is DFT (Discrete Fourier Transform), and an explanation of DFT is that a Fourier coefficient is a relativity between its sine curve and original time sequence. So it cannot avoid the interference of other periodic behaviors. Self-correlation function calculates correlation of time sequence with different time delay and cannot avoid the interference of other events happening in other time stamps. These two algorithms cannot get the start time stamp and all time stamps at which a periodic behavior happened, so we cannot know the time stamp at which the periodic behavior happened. In this paper, we proposed a multiple period acquisition algorithm which can not only mine all periodic behaviors but also get the time stamp at which a periodic behavior happened. So it can help us to predict a user's future behavior based on these periodic behaviors.

After mining the user's periodic behaviors, we have done some researches on how to recommend location based on the user's periodic behaviors. In the field of location recommendation system, some researchers have proposed some models based on the user's preference and friendships. But we found few location recommendation models based on the user's periodic behaviors during our search work. Collaborative filtering is a famous recommendation algorithm. Item CF and User CF are the basic algorithms of collaborative filtering. In this paper, because there are no friendships in our data, we use Item CF for recommending location based on the users' preference and the periodic behaviors. We designed and developed a recommendation system. This system includes an android application for showing recommending locations and collecting GPS trajectory and the server for mining periodic behaviors and recommending locations for the users based on the users' periodic behaviors.

The rest of the paper is organized as follows. Section 2 introduces the multiple periods mining algorithm which is proposed in this paper. Section 3 introduces the location recommendation model based on the periodic behaviors proposed in this paper. Section 4 introduces the thought and the architecture of our location recommendation system and we report our experimental results in Section 4. Section 5 introduces the conclusion and our future works.

2. Mining Multiple Periods in Time Sequence

In users' daily life, the user comes to a certain area not only with single period but also with multiple periods. The search of periodic behavior has attracted many researchers' attention. Recently, some algorithms proposed for mining periods are not accurate for mining multiple periods. In this section, we will introduce the proposed algorithm for mining multiple periods. For solving the problem of mining multiple periods, we proposed a new algorithm based on the thought of matrix, the thought of item hitting, and the thought of changing the matrix dynamically for reducing the execution time. Using this algorithm, we can mine all periods from time sequence.

Input: *S (time sequence)*
st (support threshold)
ts (target state)
Out put: *period (all periods in the time sequence)*
begin
(1) *matrix,map*, *T* = GeneratingMatrix (*S, ts*) // creating the suspected periods matrix based on the time-stamps the target state happened
(2) *periods* = MiningPeriods *(matrix, st,map,T)*
(3) **return** *periods*
end

Algorithm 1: Mining multiple periods.

This algorithm [17] uses the matrix for storing all suspected periods in the time sequence. In order to access elements in the matrix according to the time stamp, this algorithm constructs a hash mapping relationship between the time stamps and the number of the rows and columns of the matrix and all the items in the matrix are suspected periods. For mining true periods from the matrix, this algorithm judges all the suspected periods by counting the number of successful hitting times. If the count of hitting is more than the support threshold, the value of the current suspected period is a true period of this time sequence and if not this algorithm will judge the next suspected period by the same way. For reducing the execution time of mining all periods, this algorithm will change the value of hitting items in the matrix to be zero dynamically. This algorithm can be shown as in Algorithm 1.

In Algorithm 1, GeneratingMatrix(S, ts) is the progress of creating all suspected periods' matrix based on the time stamps at which the target state happened which will be introduced in Section 2.1. MiningPeriods(matrix, st,map,T) is the progress of mining all true periods stored in the matrix which will be introduced in Section 2.2.

2.1. Creating the Suspected Periods Stored Matrix. In the time sequence, the period of a target sate is greater than 1 and less than len(S)/2. len(S) is the length of the time sequence. If we judge whether the value from 1 to len(S)/2 is a true period, the algorithm may require a higher time complexity. Based on the characteristic that the periodic behavior always occurs at the same time interval, the period of the time series can only be the time difference between any two time stamps at which the target state happened. The algorithm of generating all suspected periods can be shown in formula (1). t_i and t_j are the time stamp at which the target state happened. sp is the suspected period. T is the time stamps at which the target state happened and L is the set of all suspected periods. This progress is shown in the following:

$$L = \left\{ \text{sp} \mid \text{sp} = t_i - t_j \left(t_i \in T \text{ and } t_j \in T \text{ and } t_i \geq t_j \right) \right\}. \quad (1)$$

Example 1. The time sequence is "0001100101010011." The target state is "1." The time stamps at which the target state happened are $\{3, 4, 7, 9, 11, 14, 15\}$. The all suspected periods can be gotten according to formula (1) as shown as $\{1, 4, 6, 8, 11, 12, 3, 5, 7, \ldots\}$. And the true periods $\{4, 5\}$ are stored in this suspected periods stored set.

In the above analysis, the time interval between any time stamps can be calculated as the suspected periods of this algorithm. While storing all suspected periods, if we choose a set or a list for storing the suspected time, the complexity of accessing the target item is $o(n)$. Therefore, this algorithm uses hash mapping algorithm and matrix to store the suspected periods and the number of the rows and the columns of the matrix mapping to the time stamps at which the target state happened. So this algorithm can access the item in the matrix according to the time stamps. And the complexity of accessing the item is $o(1)$. Another advantage of this method is that it can execute the hitting method according to the time stamps which simplify the progress of judging the suspected periods.

The progress of establishing suspected periods stored matrix can be shown in Algorithm 2. From lines (5) to (13), the algorithm acquires all time stamps at which the target state happened and creates the mapping relation to the number of rows and columns of the matrix. The hash mapping algorithm is shown in formula (2). In this formula, h is a hash value. t is the time stamp at which the target state happened. S is the time sequence. len(S) is the length of the time sequence.

$$h = t\%\text{len}(S). \quad (2)$$

From lines (15) to (22), the algorithm calculates the suspected periods according to the time stamps at which the target state happened and changes the value of the period which is bigger than len(S)/2 or less than zero to be zero for reducing the count of judging steps. In Example 2, we introduce an example for explaining the progress of Algorithm 2.

Example 2. According to the time sequence shown in Example 1, the suspected periods stored matrix can be created as

```
Input: S(time sequence)
       ts (target state)
Output: matrix (the suspected periods stored matrix)
       map (hash mapping relation)
       T (the time stamps the target state happened)
begin:
(1) map = new HashMap (key, value) // hash mapping relation between time stamps and the
number of the row and column in the matrix
(2) T = [] // the time stamps the target state happened
(3) r = 0 // the number of the row in the matrix
(4) i = 0
(5) While i < len(S) do
(6)          if S[i] == ts do
(7)                map.put (i, r)
(8)                i + +
(9)                r + +
(10)                T.append(i)
(11)         end
(12)         i + +
(13) end
(14) matrix = int[r + 1][r + 1]
(15) for t1 in T do
(16)       for t2 in T do
(17)             sp = t2–t1//suspected periods
(18)             if sp > 0 and sp < len(S)/2 do
(19)             matrix[map.get(t1)][map.get(t2)] = sp
(20)             end
(21)          end
(22) end
(23) return matrix, map, T
end
```

Algorithm 2: Generating matrix.

shown in the following based on the algorithm shown in Algorithm 2:

$$\begin{array}{c|cccccccc} & 3 & 4 & 7 & 9 & 11 & 14 & 15 \\ 3 & 0 & 1 & 4 & 6 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 5 & 7 & 0 & 0 \\ 7 & 0 & 0 & 0 & 2 & 4 & 7 & 0 \\ 9 & 0 & 0 & 0 & 0 & 2 & 5 & 6 \\ 11 & 0 & 0 & 0 & 0 & 0 & 3 & 4 \\ 14 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}. \quad (3)$$

2.2. Acquiring True Periods and Changing the Matrix Dynamically. In this step, the multiple periods mining algorithm will mine all true periods stored in the matrix step by step. The algorithm proposed in this paper will count the times the suspected period happened actually by using item hitting method. Then, the algorithm calculates the period of the target period. If the support is more than the threshold, the target period is a true period and the matrix is changed dynamically for reducing execution time. If not, the algorithm will judge the next suspected period.

During mining all true periods, the values stored in the matrix are called suspected periods. The row number of the suspected period is the first time the target state happened in this period and the column number is the second time. After mining a true period, the algorithm then changes the hitting item in the matrix to be zero. Then, the next steps will not judge the same periodic behavior. If the current item is not a true period, the algorithm will judge the next item. After judging all suspected periods stored in the matrix, the algorithm will stop and generate all mined true periods of this time sequence. The main thought of mining periods from the matrix can be shown in Algorithm 3.

In Algorithm 3, the input of the algorithm includes the storage matrix of the suspected period, the occurrence time stamp collection of the target state, the support threshold value, the hash mapping between time stamps, and the number of rows and columns of the matrix. In line (1), the algorithm initializes the period object set. Each period object contains two fields, that is, the period and the first time the periodic behavior happened. From lines (2) to (21), the algorithm mines all true periods from the matrix. Lines (9) to (15) are the progress of item hitting method. The hitting count is the times the target suspected periodic behavior happened. From lines (13) to (14), the algorithm generates the next hitting item's row and column number. If the column number

```
Input: matrix
       st
       map
       T
       S
Output: periods
Begin
(1) periods = []
(2) for t1 in T do
(3)   for t2 in T do
(4)       p = matrix[t1][t2]
(5)       If p! = 0 and p ≤ len(S)/2 do
(6)           r = t1
(7)           c = t2
(8)           hitting_count = 0
(9)           while c ≤ len(S) do
(10)                  If r in T and c in T do
(11)              hitting_count++
(12)          end
(13)          r = c
(14)          c = c + p
(15)        end
(16)      If support_sp(hitting_count, t1, p, len(S)) ≥ st do
(17)          periods.append(new PeriodObject(p, t1))
(18)          DynamicChangeMatrix(matrix, t1, t2,T)
(19)      end
(20)  end
(21) end
end
```

Algorithm 3: Mining periods.

is more than the length of time sequence, the hitting progress of the suspected periods will stop.

After executing the hitting method, the algorithm will execute the code from lines (16) to (19). Line (16) is calculating the support of the suspected period. The method of calculating support is shown in formula (4). In formula (4), SP is the support, h is the hitting count, l is the length of the time sequence, t is the time stamp at which the period happened firstly, and p is the value of the period. If the support is more than the support threshold, the algorithm will execute lines (17) to (18). In line (17), the algorithm will change the hitting items' value to be zero. Line (18) will store the mined period in the set. The method of change the matrix dynamically is shown in Algorithm 4.

$$SP = \frac{h}{(l-t)/p}. \tag{4}$$

In Algorithm 4, the method will execute the item hitting method one more time. But the difference is that in this time the algorithm will change the hitting item's value to be zero for avoiding judging the same periodic behavior more times. The full progress is shown in Algorithm 4.

In Algorithm 4, we describe the progress of the proposed multiple periods mining algorithm in detail. Example 3 will introduce the full progress according to a detailed example. In this example, we will give the situation of matrix after judging two items.

Example 3. In this example, we will continue to use the time sequence of Example 1 and the suspected periodic matrix of Example 2, while the support threshold value is 80%. (5) is the algorithm after determining whether the M[3][7] and M[3][4] are true periods. Among them, because M[3][7] is a true period, the value of the hitting element in the suspected periodic matrix is changed to 0.

The state of the matrix after judging M[3][4] and M[3][7] is as follows:

$$\begin{array}{cccccccc} & 3 & 4 & 7 & 9 & 11 & 14 & 15 \\ 3 & 0 & 1 & 0 & 6 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 5 & 7 & 0 & 0 \\ 7 & 0 & 0 & 0 & 2 & 0 & 7 & 0 \\ 9 & 0 & 0 & 0 & 0 & 2 & 5 & 6 \\ 11 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 14 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}. \tag{5}$$

The process of this example is shown as follows.

(1) For the value of M[3][4] is 1, the algorithm executes the hitting method based on the period of 1 and this periodic behavior starts from the third locations of time series. We can get that the support value of M[3][4] is 7/13. And the support

```
Input: matrix
t1
t2
p
T
Output: None
Begin
(1) While t2 ≤ max(T) do
(2)       If t1 in T and t2 in T do
(3)             matrix[t1][t2] = 0
(4)             t1 = t2
(5)             t2 = t2 + p
(6)       end
(7) end
End
```

Algorithm 4: Dynamic change matrix.

value is less than 80%. So M[3][4] is not a true period. Then, the algorithm will judge the item M[3][7].

(2) The algorithm will execute hitting method based on the period of 4 and this periodic behavior starts from the third locations of time series. We can get that the support value is 1 which is more than the support threshold. So M[3][7] is a true period. The algorithm then changes the matrix dynamically. And the state of the matrix is shown in (5). From (5), we can find that all values of the hitting items are changed to be zero. So the algorithm will not judge the same periodic behaviors. The hitting items are M[3][7], M[7][11], and M[11][15]. After this judging, the algorithm will judge the next item by the same method.

3. Location Recommendation Based on Users' Periodic Behaviors

In Section 2, we proposed a new algorithm which can mine all periods in time sequence. By using this algorithm, we can mine the user's period of arriving at a certain area. In this section, we will introduce how to recommend shops based on the user's periodic behaviors.

In the field of recommendation system, Item Collaboration Filter is a hot recommendation model. During the past several years, many companies recommended items to the users by using this algorithm. The thought of this model is that the users may buy the similar items. In this paper, we think that the users may go to the shops which are near the areas the users arrive at periodically. After mining users' periodic behaviors, we can recommend locations to the target user according the period and the latitude and longitude of the periodic arrival area. We choose the Item CF algorithm for calculating the user's preference of a certain location. For recommending locations to the user personally, in this paper, we choose the user's taste and the distance between the shop and the center of the periodic arrival area as this model's characteristics.

According to the thought of item-based collaborative filtering algorithm, in this paper, we use cosine-based similarity algorithm to calculate the similarity between two items. The formula is shown in formula (6). i and j are two shops' feature vectors. In this paper the feature vector is (*similarity_of_user's_taste, the_distace*). The first element of this vector is the similarity between the shop's tag and the user's taste. The second element of this vector is the distance of this shop to the center of this user's periodic arrival area:

$$\operatorname{sim}(i, j) = \cos\left(\vec{i}, \vec{j}\right) = \frac{\vec{i}.\vec{j}}{\left\|\vec{i}\right\|_2 * \left\|\vec{j}\right\|_2}. \tag{6}$$

After calculating the similarity of the shops, in this paper, we calculate the user's preference of a shop by using formula (7). In this formula, $S_{i,N}$ is the similarity between i and N. And $R_{u,N}$ is the user's preference of the shop N that this user has been there.

$$P_{u,i} = \frac{\sum_{\text{all\_similar\_items},N}\left(s_{i,N} * R_{u,N}\right)}{\sum_{\text{all\_similar\_items},N}\left(\left|s_{i,N}\right|\right)}. \tag{7}$$

After calculating the user's preference of these shops, we can sort these shops according to preference. And then we can give this user some suggestions for deciding which shop this user can go to. In real life, we may go to the same shop for shopping or having a dinner. In our project, we do not filter out the shops this user has been to. In Section 4, we will introduce the system we developed based on this recommendation model.

4. Experiments and System

In this section, we will use the algorithm proposed in this chapter to conduct multiple periodic behavior mining experiment. We use the check-in data collected in a company. In this section, we give the comparative experiment analysis of the multiple periods mining algorithm proposed in this paper. Then, we will introduce the shops recommendation systems based on the user's periodic behaviors.

4.1. Comparative Experiment Analysis of the Multiple Periods Mining Algorithm. In this section, we mined periodic behaviors by using periodogram and the algorithm proposed in this paper. The time sequence is shown in (8). The result of periodogram algorithm is shown in Figure 1. In this experiment, we set the support threshold to be 80%. From the original time sequence, we can find that this user has two periodic behaviors directly. The employees come to this company every week on Friday and every two weeks on Tuesday.

The time sequence is as follows:

$$\begin{aligned}&0,1,0,0,1,0,0,0,0,0,0,1,1,1,0,1,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0,0,0,\\&0,1,0,0,0,1,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0.\end{aligned} \tag{8}$$

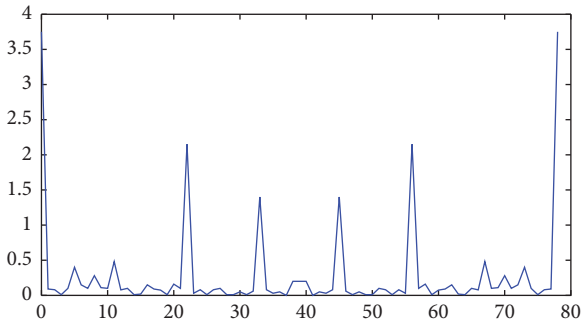


Figure 1: The power spectrum of periodogram.

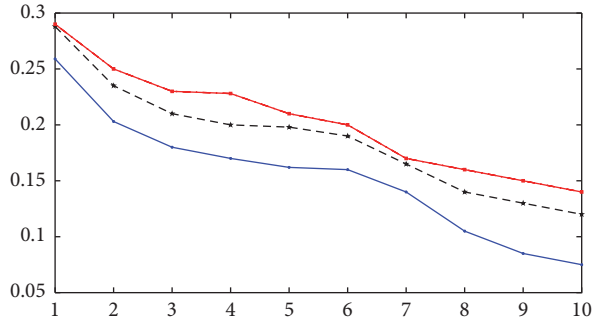


Figure 2: The result of recommendation experiment.

Table 1: The result of the proposed algorithm.

Periods	Time stamps
7	2,16,30,44,72,
14	5,12,26,33,40,47,54,61,68,75,

Firstly, we mine multiple periodic behaviors by using the proposed algorithm in this paper. The result can be seen in Table 1. We can find that the result of this experiment is correct.

Then, we mine the periodic behaviors by using the algorithm of periodogram. The result of density of power spectrum is shown in Figure 1. The x-axis is the frequency and the y-axis is the density of power spectrum. We take the frequency of the power spectrum in the top as the true frequency. According to formula (9), we can calculate the period. Because the period is an integer, we get the period of 7. In formula (9), len(s) means the length of the time sequence.

$$\text{period} = \frac{\text{len}(S)}{\text{frequency}}. \tag{9}$$

In this experiment, we found that the periodogram cannot mine all true periods because of spectral leakage. The period of 14 cannot be divided by 77 (the length of this time sequence). So when we mine periodic behavior by periodogram, we cannot get the period of 14 directly. And the periodogram cannot tell us when the behavior happens in a period. Based on this period, we cannot predict the user's behavior in future. The proposed algorithm in this paper solved this problem.

4.2. Location Recommendation Experiment. In our location recommendation experiment, we use the GPS trajectory collected in our system as our experiment's data. We mine the user's periodic behavior by using the method proposed in Section 2. Based on the periodic behaviors, we generate the recommending location list by using the method proposed in Section 3. In this section, we will use the accuracy to evaluate the recommendation method proposed in this paper. The accuracy of our recommendation method is shown in Figure 2. The x-axis of the figure is the count of recommending locations. And the y-axis of this figure is the accuracy. The red line is the result of the algorithm proposed in this paper. The black line is the result of item-based CF algorithm and the blue line is the result of the user-based CF algorithm. In Figure 2, we evaluate the item-based CF algorithm without using the characteristic of periodic behaviors. We can see that the recommendation method proposed in this paper is higher than the raw item-based CF algorithm. We also recommend the location to the user by using user-based CF algorithm and it is less accurate than our model. From this result, we can know that mining users' behavior is important for recommending the users locations, because the user may always go a location which is near his frequent or periodic arrival area. For example, we may go shopping near our home, go to a restaurant for a dinner near our working place, and so on. We give an example of the result of location recommendation as follows.

The list of recommending shops is as follows:

Locations List

(1) KFC

(2) Green Tea

(3) Northeast Restaurant

(4) Boiled dumplings

⋮

4.3. The Location Recommendation System Based on the User's Periodic Behaviors. In our system, we developed the mobile application and the server-side application. In the mobile side, the application includes GPS trajectory collection module and the location recommendation module. In the server side, the application includes the GPS trajectory storage module, the stay-points mining module, the frequent arrival area mining module, and the periodic behavior mining module. For recommending location to the users, we also developed the location recommendation module based on the user's periodic behaviors in the server side. In Figure 3, the architecture of our system can be seen. We divided the system into three layers. They are the mobile application layer, the service providing layer, and the data storage layer. The mobile application layer can access the service providing layer through HTTP and the service providing layer can access the data storage layer through JDBC technology. In this section, we will introduce the data processing flow of this system in Section 4.3.1.

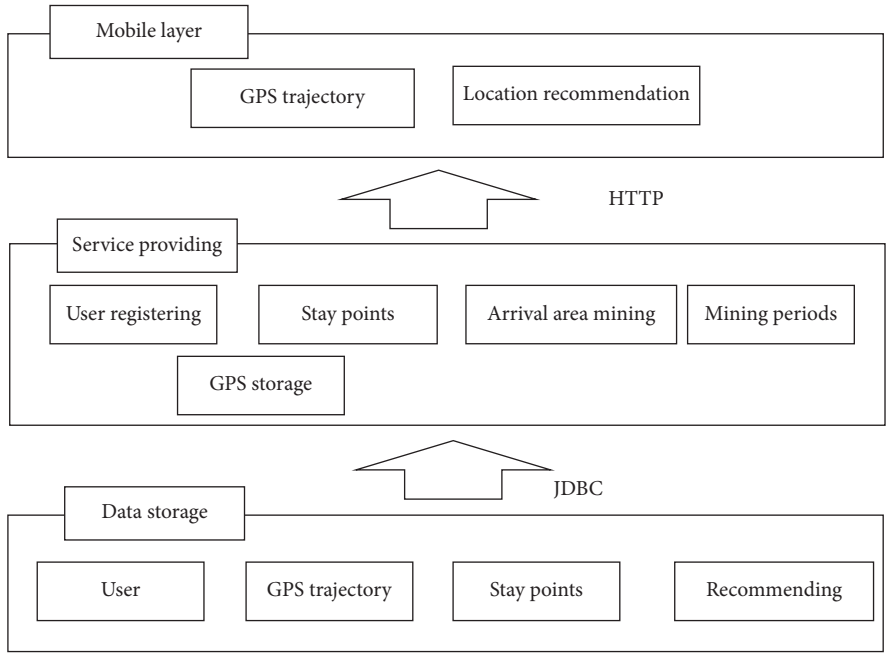


Figure 3: The architecture of our system.

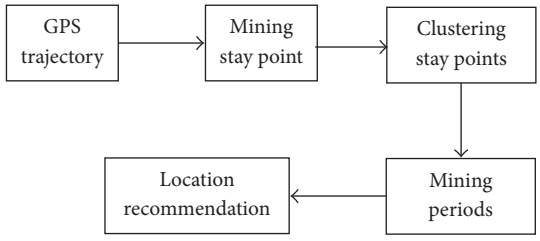


Figure 4: The data processing flow of our system.

4.3.1. The Data Processing Flow of This System. In our system, we design the data processing flow as shown in Figure 4. The server-side application receives the GPS data from the mobile client and stores the data in database. Then, it mines the stay points from the raw GPS data of each user, because a stay point means a staying behavior of a user and the GPS positions of all stay points in a frequent arrival area are not the same. The server then mines semantic area by using OPTICS [21] clustering algorithm. The OPTICS algorithm is a kind of the clustering algorithms based on the density of data. Because it does not need to set the number of classes and we cannot understand the number of the user's frequent arrival area, we choose it for mining the user's frequent arrival area. Such as the K-Means clustering algorithm, we should give the number of classes firstly and then the algorithm can mine all classes effectively; it is not suitable for our application. The OPTICS clustering algorithm is used in many areas widely such as web clustering. Based on the frequent arrival areas mined by the OPTICS algorithm, the server-side application mines the periods of each frequent arrival area by using the periods mining algorithm proposed in this paper.

After mining the user's periodic arrival areas and the periods, we designed the location recommendation module based on the user's periodic behaviors. For each periodic arrival area, we generated the recommending location list by using item-based CF algorithm. In the server side, the system can predict the user's future arrival area based on the periodic behavior. After predicting the future arrival area, this system pushes the recommending location list to the mobile application. The user can consider these locations as a future choice. As introduced above, the architecture of the data processing flow can be seen in Figure 4.

5. Conclusion and Future Works

In this paper, we proposed a new periods mining algorithm which can mine all periods in the time sequence and proposed a periodic behaviors based recommendation method for recommending the locations to the users. From this paper, we can see that our periods mining algorithm is more accurate than some other algorithms and the recommending model is more accurate than the raw item-based CF algorithm. But in this paper we do not consider the effect of the user's friendship in our recommending methods. And we found that the group's periodic behavior is also a true phenomenon in our daily life. So in the future, we will go on doing some research in a location recommending methods based on the friendship and the user's periodic behaviors. And we will do some research on how to mine the group's periodic behaviors.

Disclosure

This paper is extended from the paper named "Mining Multiple Periods in Event Time Sequence" in APSCC 2015.

Competing Interests

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

Acknowledgments

This work is partially supported by the National Natural Science Funds of China under Grants no. 61173042 and no. 61472004, Hong Kong, Macao, and Taiwan Science and Technology Cooperation Program of China under Grant no. 2013DFM10100, and Special Fund Project of Shanghai Economic and Information Committee under Grant no. CXY-2013-40.

References

[1] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pp. 325–334, ACM, Beijing, China, July 2011.

[2] A. Papadimitriou, P. Symeonidis, and Y. Manolopoulos, "Friendlink: link prediction in social networks via bounded local path traversal," in *Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN '11)*, pp. 66–71, Salamanca, Spain, October 2011.

[3] V. Bellotti, B. Begole, E. H. Chi et al., "Activity-based serendipitous recommendations with the magitti mobile leisure guide," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, pp. 1157–1166, ACM, Florence, Italy, April 2008.

[4] R. Lee, S. Wakamiya, and K. Sumiya, "Discovery of unusual regional social activities using geo-tagged microblogs," *World Wide Web*, vol. 14, no. 4, pp. 321–349, 2011.

[5] H. Gao, J. Tang, and H. Liu, "Addressing the cold-start problem in location recommendation using geo-social correlations," *Data Mining and Knowledge Discovery*, vol. 29, no. 2, pp. 299–323, 2015.

[6] J.-D. Zhang, C.-Y. Chow, and Y. Li, "IGeoRec: a personalized and efficient geographical location recommendation framework," *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 701–714, 2015.

[7] H. Huang, "Context-aware location recommendation using geotagged photos in social media," *ISPRS International Journal of Geo-Information*, vol. 5, no. 12, p. 195, 2016.

[8] Z. Yuan and H. Li, "Location recommendation algorithm based on temporal and geographical similarity in location-based social networks," in *Proceedings of the 12th World Congress on Intelligent Control and Automation (WCICA '12)*, pp. 1697–1702, Guilin, China, June 2016.

[9] J. Rekimoto, T. Miyaki, and T. Ishizawa, "LifeTag: WiFi-based continuous location logging for life pattern analysis," in *Location- and Context-Awareness: Third International Symposium, LoCA 2007, Oberpfaffenhofen, Germany, September 20-21, 2007. Proceedings*, Lecture Notes in Computer Science, pp. 35–49, Springer, Berlin, Germany, 2007.

[10] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie, "Mining individual life pattern based on location history," in *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM '09)*, pp. 1–10, May 2009.

[11] M. Lahiri and T. Y. Berger-Wolf, "Mining periodic behavior in dynamic social networks," in *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM '08)*, pp. 373–382, December 2008.

[12] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 1099–1108, ACM, Washington, DC, USA, July 2010.

[13] M. S. Bartlett, "Periodogram analysis and continuous spectra," *Biometrika*, vol. 37, pp. 1–16, 1950.

[14] Z. Wang, "Analysis of the hydrological cycle in the middle and upper reaches of the Yellow River," *Northwest Hydropower*, no. 2, pp. 1–5, 1998.

[15] X. Tao, "Preliminary study of periodic regularity of scarlet fever by periodogram method," *Journal of Preventive Medicine*, vol. 14, no. 3, pp. 146–148, 1998.

[16] X. Wei, "Window function analysis of power spectrum estimation by periodic graph method," *Modern Electronic Technology*, vol. 28, no. 3, pp. 14–15, 2005.

[17] Y. Hua, R. Yang, and Y. Qian, "On detecting customer behavior periodicity with cross entropy," in *Proceedings of the ICIS SIGBPS Workshop on Business Processes and Services (BPS '12)*, pp. 3–7, 2012.

[18] S. Parthasarathy, S. Mehta, and S. Srinivasan, "Robust periodicity detection algorithms," in *Proceedings of the 15th ACM Conference on Information and Knowledge Management (CIKM '06)*, pp. 874–875, Arlington, Va, USA, November 2006.

[19] H. Wang, D. Liu, C. Wang et al., "Wavelet periodic analysis model of hydrological series based on seasonal adjustment and trend decomposition and its application," *Journal of Applied Science and Engineering*, vol. 21, no. 5, pp. 823–836, 2013.

[20] B. Xu, Z. Ding, and H. Chen, "Mining multiple periods in event time sequence," in *Advances in Services Computing*, vol. 9464 of *Lecture Notes in Computer Science*, pp. 278–288, Springer, 2015.

[21] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '99)*, pp. 49–60, ACM Press, Philadelphia, Pa, USA, June 1999.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 7653706, 11 pages
https://doi.org/10.1155/2017/7653706

Research Article

Recovering Individual's Commute Routes Based on Mobile Phone Data

Xin Song,[1] Yuanxin Ouyang,[1] Bowen Du,[1] Jingyuan Wang,[1] and Zhang Xiong[1,2]

[1]*School of Computer Science and Technology, Beihang University, Beijing, China*
[2]*Research Institute of Beihang University in Shenzhen, Shenzhen, China*

Correspondence should be addressed to Bowen Du; dubowen@buaa.edu.cn

Received 20 September 2016; Revised 17 November 2016; Accepted 12 December 2016; Published 9 February 2017

Academic Editor: Qingchen Zhang

Mining individuals' commute routes has been a hot spot in recent researches. Besides the significant impact on human mobility analysis, it is quite important in lots of fields, such as traffic flow analysis, urban planning, and path recommendation. Common ways to obtain these pieces of information are mostly based on the questionnaires, which have many disadvantages such as high manpower cost, low accuracy, and low sampling rate. To overcome these problems, we propose a commute routes recovering model to recover individuals' commute routes based on passively generated mobile phone data. The challenges of the model lie in the low sampling rate of signal records and low precision of location information from mobile phone data. To address these challenges, our model applies two main modules. The first is data preprocessing module, which extracts commute trajectories from raw dataset and formats the road network into a better modality. The second module combines two kinds of information together and generates the commute route with the highest possibility. To evaluate the effectiveness of our method, we evaluate the results in two ways, which are path score evaluation and evaluation based on visualization. Experimental results have shown better performance of our method than the compared method.

1. Introduction

Human mobility has been a significant research area in recent years. An improved understanding of human mobility is meaningful in lots of fields, such as predicting the spread of disease, evaluating the effect of human travel on the environment, and urban planning.

Common questionnaire based ways to obtain the information of human mobility have quite a lot of flaws. Dealing with questionnaires can cost huge amount of manpower and spend quite a lot of time. Due to the huge cost of manpower and time, the number of samples is limited. Another problem is that people are easily affected by subjective factors which can make the result of the questionnaire unstable. To address all these problems, there are plenty of researches studying human mobilities based on GPS data [1] and taxi location trajectories data [2]. But these data have one big flaw, which is that they cannot cover the people wildly enough. Differently, mobile phone users have a wild coverage not only in developed countries but also in developing countries. Besides, mobile phone data contain rich information that can be used in multiple domains, such as revealing people's travel trajectory [3, 4], mining important locations [5], finding the spatial nature of human mobility [6], assisting the demographic census [7], and studying communication network [8].

Due to the wide coverage and tremendous information embedded, mobile phone data are quite suitable for analysing human mobilities. Individual's commute route is an important part of human mobility. Specifically, knowing people's commute routes has great importance in terms of at least three conspicuous aspects: (1) traffic flow analysis: we can infer each road segment's level of congestion from multiple people's commute routes; (2) urban planning: after the new road is mended, we can observe whether there is a change of individuals' commute routes so as to judge the effectiveness of the new mended road; (3) personalized services: according to many other people's common commute routes, individuals can be recommended commute routes and way of transportation based on their home and work places.

We are facing three main challenges. (1) The first is the low precision of location information embedded in the data. We can only use the cell towers coordinates to approximately represent people's real history locations. And all the cell phones within the distance of 1000 meters from the tower can receive its signal. So this will cause the low precision of location information. (2) The second challenge is that the time interval between two adjacent signal records of one mobile device can be quite long, which sometimes can reach one hour. So the sampling rate of the location trajectory is extremely low. (3) The third challenge is from the fact that people may have multiple commute paths. Changing the transportation tools always means changing the route, so finding the most possible path from all the overlapping everyday commute paths is our last challenge.

Facing all these challenges, we propose a commute routes recovering model to recover individual's commute route from mobile phone data. The model includes two main modules: data preprocessing module and map matching module. The first module extracts commute trajectories from raw dataset and formats the road network into a better modality. The second module combines two kinds of information together and generates the commute route with highest possibility. To the best of our knowledge, this is the first work focusing on recovering commute routes through mobile phone data of multidays. On the whole, this paper offers the following contributions:

(i) We design a data preprocessing module to form the data into suitable modalities for the route recovering task. For the mobile phone data, we apply the leader clustering algorithm to cluster the nearby cell towers and adopt an important location detecting strategy to find individual's home and work place. For the road network data, we design one road segmentation algorithm and one road merging algorithm to format the road segments.

(ii) We design the map matching module which firstly fuses the trajectory information extracted from mobile phone data with real world road network data and then adopts a path generating algorithm to generate the path with highest possibility.

(iii) To evaluate the effectiveness of the proposed model, we design two evaluating methods: path score evaluation and evaluation based on visualization. The path score evaluation calculates one score for each path, which considers the number of nearby cell towers of each path. Then we design the visualizing part drawing all the related data on the map to show the whole process of path recovering. This can directly and clearly show the relevance between the raw trajectory data and the generated commute path and can prove the better performance of our model.

The rest of this paper is structured as follows. Section 2 reviews the related work. Section 3 describes the mobile phone data and the real world road network data we used in this paper. We introduce the whole framework of the proposed method for recovering individual's commute routes in Section 4. Section 5 shows the experimental results and visualization of all the commute information. And the paper is concluded in Section 6 with a brief discussion of limitations and directions of future research.

2. Related Work

2.1. Application of Mobile Phone Data. Applications of mobile phone data have been a hot spot of research areas in recent years, which is mainly due to the wild coverage of mobile devices among people. Besides, there is rich information embedded in the mobile phone records which can be used in multiple domains, such as revealing peoples travel trajectory [3, 4], mining important locations [5], finding the spatial nature of human mobility [6], assisting the demographic census [7], protecting the identity, location, and sensitive information [9], and studying communication network [8].

Researches on human mobilities mainly focus on mining peoples mobility patterns [10, 11] and identifying important locations [12]. Differently, our work focuses on the specific commute routes of people which is quite meaningful in transportation-related areas.

2.2. Multimodal Data Fusion. With the era of big data coming, multiple kinds of data have been generated in different domains. Researchers from all over the world try to solve problems based on various data. Data from different domains always have multiple modalities, each of which has a different representation, distribution, scale, and density [13].

To find the traffic regularity between city areas, Zheng et al. [14] adopted a two-stage model, which firstly partitions a city into regions by major roads using map segmentation method [15] and then maps the GPS trajectories of taxicabs onto the regions to formulate a region graph. DNN-based model can be used to learn new feature representations through data with same modality [16, 17] and data with different modalities [18, 19] while concerning data privacy [20, 21]. Xin et al. [22] present a multisource active transfer learning framework for entity resolution task. Blum and Mitchell [23] employ cotraining method using a large unlabeled sample to boost performance of a learning algorithm when only a small set of labeled examples is available. Wang et al. [24] fuse multiple features in face recognition task. A new method for multiview dimensionality reduction is proposed by Zhang et al. [25]. Zhang et al. cluster incomplete multimedia data based on tensor distance [26, 27]. Rong et al. [28] utilise association rules to add group information to personal profiles.

In our work, we fuse cell tower location trajectories with real world road network data by the transfer matrix defined in our proposed model, which is different with existing approaches.

2.3. Map Matching. Map matching problem refers to the task of matching a raw trajectory to roads on a digital map. Map matching algorithms can be categorized into local/incremental algorithms [29] and global algorithms [30] according to the range of sampling points considered when matching the trajectories [31].

Yuan et al. [32] propose an Interactive Voting-Based Map Matching algorithm to solve the problem of low sampling rate GPS trajectories. GPS signal is recorded no longer than every 2 minutes, but the time interval between two mobile phone records can be nearly half an hour. And considering the low precision of location information, traditional methods cannot be used on the mobile phone data.

Thiagarajan et al. [33] propose an energy-efficient system for trajectory mapping using raw position tracks obtained largely from cellular base station fingerprints. The mobile phone data used in their work have a much higher signal sampling rate which are different from the data recorded by the real mobile operators, so they could reconstruct the route just based on one day trajectory. Our work is based on the real world mobile phone data which means the sampling rate is extremely low, so we combine the trajectories in multidays to increase transfer information. To the best of our knowledge, our work is the first to recover people's commute routes based on multidays' mobile phone data.

3. Data Description

In this section, we introduce two datasets used in this paper, which are mobile phone dataset and road network dataset. We extract individuals commute trajectories in multiple days from mobile phone dataset. The road network dataset is used to map the cell tower trajectories to real world road paths.

3.1. Mobile Phone Dataset. The mobile phone data used in this paper were collected during the period from October 24, 2013, to March 24, 2014, in Wuxi, China, containing about six million users equally spread over space. All the users can totally generate 40 million raw records each hour everyday which include huge amount of location information recorded in form of cell-id, area-id which can singly represent one cell tower. Based on the geographical data which contain the coordinate of each cell tower, we can easily transfer the cell tower id into coordinates. Each record in the raw dataset contains four parts: user id, cell tower id, time stamp, and tag. The time stamp can record the precise time when this record was recorded. The tag shows the specific activity one record stands for. The records are generated when the users are engaged in communication via the cellular network. Specifically, the records are recorded at the beginning and the end of each voice call placed or received, when a short message is sent or received, and when Internet is connected. So the cell tower's coordinate can approximately represent the user's history locations.

To better overcome the challenge of low sampling rate in our experiment, we tend to choose the devices that can generate stable and adequate data. The rule is that the chosen devices need to be recorded at least 24 records in each day during the selected period. Besides, we remove the individual's data whose home and work place are the same.

Administrative region of Wuxi includes three main parts: two small towns and one larger city. Figure 1 shows the spatial distribution of cell towers in Wuxi; as we can see, urban area has higher density of towers than the suburb area.

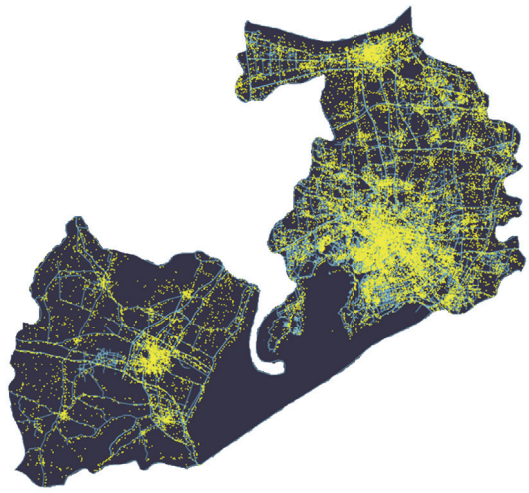


Figure 1: Spatial distribution of cell towers.

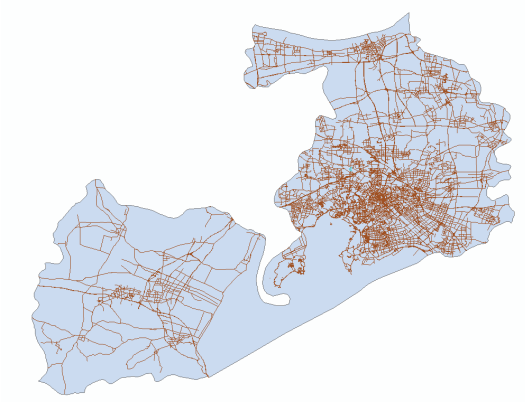


Figure 2: Real world road network in Wuxi.

3.2. Road Network Dataset. The road network dataset used in our work contains real world road information in Wuxi. Figure 2 shows the spatial distribution of all the road segments. Similarly, there are much more road segments in urban areas than the suburb. Each road segment in the dataset includes multiple points which are sampled from the corresponding real world road. The length of each raw road segment varies a lot. And there are 12158 road segments in the dataset. Based on the dataset we can take the whole information of real world roads into consideration when generating the commute routes of individuals.

Due to the information privacy concern, all the data are anonymous; we note that no private data are used in the experiment.

4. Model

As is shown in Figure 3, the framework of commute route recovering model contains two main modules: data preprocessing module and map matching module.

Data preprocessing is the first step of the proposed method for recovering individuals commute route. Data

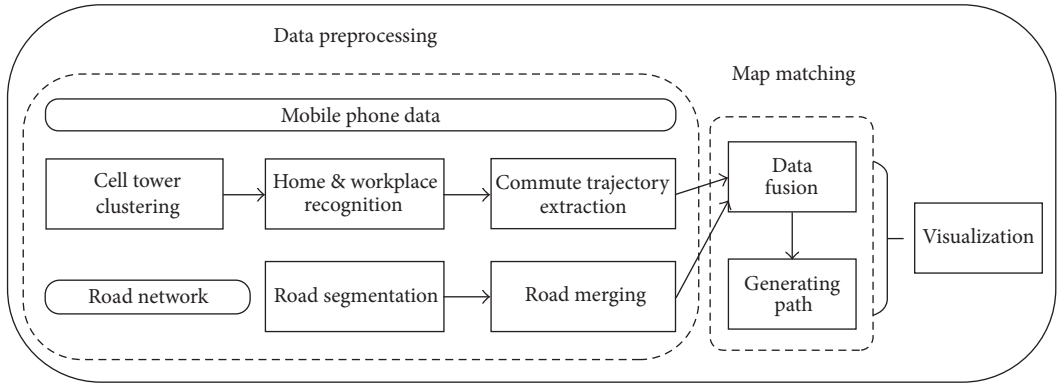


Figure 3: Framework of commute route recovering model.

preprocessing module contains two main parts: trajectory extraction and road network formation. Trajectory extraction submodule includes three steps: raw data clustering, important location detection, and commute trajectory extraction, aiming to extract individuals' everyday paths from home to work place in form of cell group trajectories. Then the road network formation submodule contains two steps: road segmentation and road merging. This module aims to reconstruct the raw road network data into suitable formation which can reduce the calculating time and increase the accuracy of the result.

Map matching module includes two steps. The first step is mapping commute trajectories to the real world road segments so as to generate the road segments transfer matrix, which is the key point for multidays data fusion task. And the second step is generating the commute paths in form of continuous road segments trajectories based on the transfer matrix.

4.1. Trajectory Extraction. To recover individual's commute route which specifically refers to the path between home and work place, we firstly extract individual's history location trajectories from raw mobile phone dataset. As is mentioned above, each location point in the trajectories represents one cell tower id which can be transferred to its corresponding coordinate (longitude, latitude). Let seq_i denote a sequence of records of user i in one day such as $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$, where l_k^i is the kth location of user i. This is the raw trajectory that we can easily obtain from the raw dataset. Then the module aims to transfer $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$ *to* $seq_i = \{g_1^i, g_2^i, \ldots, g_n^i\}$, where g_k^i represents the kth group of several nearby locations which is generated by the clustering step and g_1^i represents the location of home and g_n^i represents the location of work place which are detected by the important location detection module.

4.1.1. Leader Clustering. Figure 4 shows one person's history locations in multiple days. Each yellow point represents one cell tower and the size of the point is proportional to the number of records recorded by the corresponding cell tower, which means the bigger the point is, the more possible one person will appear at that area. In the real world, a motionless mobile phone device may contact with different cell towers at different time and the cell tower reselection often happens where cell towers' coverage overlaps with others, so two nearby points in the map may be generated by users in the one single place and should be clustered in one single group.

Input: all the points of one person, denote as P
Output: all the groups of one person, denote as G;
(1) **while** P.size! = 0 **do**
(2) P_{leader} = SelectLeader(P)
(3) P.remove(P_{leader})
(4) *nearbyPoints* = selectNearbyPoints(P)
(5) $group_i$.addAll(*nearbyPoints*)
(6) P.removeAll(*nearbyPoints*)
(7) **end while**

Algorithm 1: Leader clustering algorithm.

We apply leader clustering algorithm [34] to cluster nearby cell towers into corresponding groups to handle the cell tower reselection problem. The reason we choose leader clustering algorithm is that it does not require the clusters' number before clustering but needs a weight for each point so as to pay more attention to the leader points, which is exactly suitable for this problem. As is mentioned above, each point has one value which represents the number of records recorded by the corresponding cell tower and we use this value as the weight of each point. The time complexity of leader clustering algorithm is $O(N_p \times k)$, where N_p represents the number of all points and k refers to the total number of clusters and the space complexity is $O(N_p)$.

As is shown in Algorithm 1, we firstly select one leader among all the points that have not been clustered based on the weight of each point; then we put all the nearby points which are within the distance of 300 meters from the leader into one group. We keep doing this procedure until all the points have been grouped. Figure 4 shows all the groups of one person's history locations, each member of the group is connected by lines, and the leader of the group is covered by a blue circle.

Figure 4: Cell groups after leader clustering.

Choosing a suitable radius in the clustering process is necessary. But the difficulty is that different areas have different number of cell towers; for example, urban area contains more towers than the suburban area. The average distance between each pair of cell towers is less than 1 km. We tried a range of radius to do the clustering and found that 300 m performs well in our experiment.

4.1.2. Important Location Detection. To extract the cell group trajectory from home to work, we firstly need to find out where home and workplace is. Intuitively we believe that people will stay at home and workplace much longer than other places, so there should be more records recorded around these important places. After the clustering, we treat the group of cell towers as the smallest unit representing individual's history locations. To find out people's home and workplace, we adopt a simple but useful strategy. As mentioned by Isaacman et al. [12], most of people spend the leisure time between 7 p.m. and 6 a.m. at home and spend the work time between 1 p.m. and 5 p.m. at workplaces. So we calculate R_i for each group which represents the total number of records recorded by all the cell towers in the group during specific period. Home is detected as follows:

$$home = \{group_i \mid \max(R_i) \cap t \subset HomeTime\}. \quad (1)$$

Similarly, workplace is selected as follows:

$$work = \{group_i \mid \max(R_i) \cap t \subset WorkTime\}. \quad (2)$$

4.1.3. Commute Trajectory Extraction. Through clustering step, we can transfer individual's everyday sequence $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$ *to* $seq_i = \{g_1^i, g_2^i, \ldots, g_n^i\}$, where l_i refers to locations of cell towers and g_i refers to groups of cell towers. Based on the important location detection step, we can obtain the id of groups which are around home or workplace; then we capture the trajectory between home and workplace.

We used four weeks' data in our experiment, and most of the people have 20 trajectories from home to workplace during the four weeks' period. As is shown in Figure 5, each colored line represents one trajectory from home to workplace.

After extracting all the trajectories, we form the group transfer matrix $M_{G\times G}$ based on these trajectories. Each element M_{ij} in the matrix records the frequency for the ith group transfer to the jth group in all the trajectories. Matrix $M_{G\times G}$ is the key point for the multiday data fusion task, aiming to increase the sampling rate of the commute path.

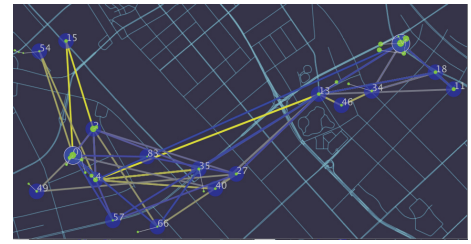


Figure 5: Commute trajectories in multiple days.

4.2. Road Network Formation. Road network dataset is used to transfer cell groups trajectories to the real world road segment trajectories. Road network formation module reconstructs raw road network data with a better modality. One road segment in the raw dataset is stored as a series of points. For example, road segment $R_i = \{p_1, p_2, \ldots, p_n\}$, where p_i refers to the ith point of the road segment.

In this subsection, we introduce two operations for the road network dataset, road segmentation, and road merging, which can increase the precision of the generated path and reduce the time complexity of the algorithm.

4.2.1. Road Segmentation. As is shown in Figure 6, there are some extremely long road segments that have multiple common points with other road segments. This will decrease the precision of map matching procedure because we treat each road segment as the smallest unit. Road segmentation step aims to divide the long road segment into several short segments which contain no common points with other road segments inside the road. For example, we divide road segment $R_i = \{p_1, p_2, \ldots, p_k, \ldots, p_n\}$ into two parts: $R_i^1 = \{p_1, p_2, \ldots, p_k\}$ and $R_i^2 = \{p_k, \ldots, p_n\}$, where p_k is an intersection point between R_i and other segments.

Segmentation procedure contains two steps. The first step is finding out all the common points between each pair of road segments. The second step is dividing long road segments into several short segments based on the common points. After this step, each road segment contains no exit point between the start and end points.

4.2.2. Road Merging. Figure 7 shows another flaw of the raw dataset. As we can see, each road segment has at least one common point with other segments. The problem is that there are some really short road segments that can totally be attached to their adjacent longer road segments. This can reduce the total number of road segments and the time complexity of this step. The problem actually lies in the raw dataset; road segmentation step cannot cause this problem.

To deal with this, we apply an algorithm which firstly finds the all the common points that exactly belong to two road segments. Then it merges the corresponding road segments

Figure 6: Long road segment.

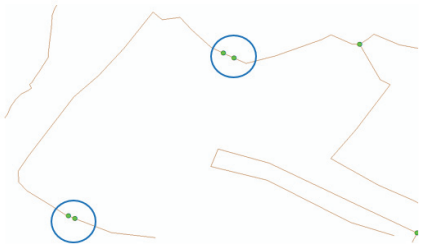


Figure 7: Redundant road segment.

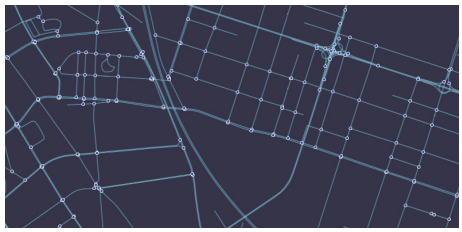


Figure 8: Road network after formation.

based on the common points. Algorithm 2 shows the detailed procedure of road merging algorithm. And Figure 8 shows the road network after the formatting step. The time complexity of the algorithm is $O(N_r)$, where N_r represents the number of road segments and the space complexity is $O(N_p)$, where N_p refers to the number of sampling points consisting of all the road segments.

4.3. Map Matching. In this subsection, we will introduce the procedure of map matching module which includes two parts: data fusion and path generation. The map matching module aims to fuse extracted cell group trajectories with real world road network so as to recover individual's commute route. Data fusion step calculates the road segment transfer matrix $M_{G\times G}$; then path generating step recovers the most likely commute route based on the matrix $M_{G\times G}$.

4.3.1. Data Fusion. This step aims to map each group in the commute cell group trajectories to the corresponding road segments and then form the road segments transfer matrix $M_{R\times R}$ based on the frequency recorded in matrix $M_{G\times G}$. Given matrix $M_{G\times G}$, Algorithm 3 shows the detailed procedure of the data fusion algorithm.

Input: raw road network R_{raw}
Output: format road network R_{format};
(1) cp = findCommonPoints();
(2) $pTwo$ = CalculatiPointsOnlyBelongToTwoRoads(cp);
(3) **for** *point* in $pTwo$ **do**
(4) mergeRoad(*point*);
(5) **end for**

Algorithm 2: Road merging algorithm.

Input: cell group transfer matrix: $M_{G\times G}$
Output: road segment transfer matrix: $M_{R\times R}$;
(1) **for** m_{ij} in $M_{G\times G}$ **do**
(2) $pointSet_i$,$pointSet_j$ = GetAllMembers(i, j);
(3) **for** p_m in $pointSet_i$ **do**
(4) **for** p_n in $pointSet_j$ **do**
(5) $weight = m_{ij}\times$ Weight(p_m) $\times$ Weight(p_n);
(6) $road_a$,$road_b$ = GetNearestRoad(p_m, p_n);
(7) $M_{R\times R}[road_a][road_b]$ += $weight$;
(8) **end for**
(9) **end for**
(10) **end for**
(11) $M_{R\times R}$ = MakingContinuousByDijkstra($M_{R\times R}$);

Algorithm 3: Map matching algorithm.

The transfer frequencies of all pairs of cell groups are recorded in the matrix $M_{G\times G}$. One cell group contains multiple location points; we select the road segment set for each group which includes all the nearest road segments for each point in the group. Then the transfer frequency of two groups is distributed to all pairs of road segments from one set to another according to the weight of the corresponding points in the group.

Through fusing multiday data, we hugely increased the number of location records around individual's commute route. But there still exit some adjacent recorded points whose corresponding road segments are not contiguous with each other. This can cause the discontinuity of the path. To solve this problem, we adopt Dijkstra algorithm to complete each pair of road segments with the shortest path between them. Because as the number of records is increasing, the average distance between two adjacent recorded points will become much smaller. And in most of the cases, people will choose the shortest path when passing two enough close road segments.

The complexity of map matching algorithm is acceptable. Let N_g represent the number of cell groups for one person. Most of the people passed less than 100 cell groups. Let n_p represent the average size of all groups. And the value of n_p is less than 10. Then the time complexity of the algorithm is $O(N_g \times n_p^2)$, and the space complexity is $O(N_r^2)$, where N_r refers to the number of all the road segments contained in the road segment set mentioned above which is quite a little part of the whole road network.

Input: road to road transfer matrix: $M_{R\times R}$
Output: the most possible path: $P = \{r_1^i, r_2^i, \ldots, r_n^i\}$;
(1) $maxFrequence$ = findMaxFrequence($M_{R\times R}$);
(2) **for** m_{ij} in $M_{R\times R}$ **do**
(3) if(m_{ij}==0)
(4) m_{ij} = Double.Max_Value;
(5) else
(6) $m_{ij} = (maxFrequence - m_{ij})/m_{ij}$;
(7) **end for**
(8) P = Dijkstra($M_{R\times R}$, R_{begin}, R_{end})

Algorithm 4: Path generating algorithm.

4.3.2. Path Generation. The path generating module is the final part of the whole model, which generates the most likely commute route for each person. The final commute route is a trajectory of continuous road segments. Given road segments transfer matrix $M_{R\times R}$, Algorithm 4 shows the detailed procedure of route recovering algorithm.

A path's frequency is equal to the sum of all its contained road segments' frequency. And commonly we believe that the path with the highest frequency may be the most possible commute route, but the problem is that the longest path will definitely have the highest frequency among all, and the most possible path should be a relatively shorter one. To balance the length and the total frequency of the path, we recalculate the value of each element m_{ij} in matrix $M_{R\times R}$ according to the following formula:

$$m_{i,j} = \frac{maxValue - m_{i,j}}{m_{i,j}}, \tag{3}$$

where m_{ij} represents the element of the road transfer matrix and $maxValue$ represents the max value in the matrix. So, for example, the road segment with the highest frequency will get zero for the new value and contrarily the road segment with lower frequency will get a higher value. Plenty of formulas have been tried and we found that the formula above performs well in our experiment. Finally, we adopt Dijkstra algorithm to generate the commute path. The time complexity of this algorithm is $O(N_r^2)$.

The entire process of path recovering for one individual takes 1083 milliseconds and uses nearly 1 G of memory. One thing that needs to be noted is that finding the shortest path between two road segments usually costs quite a lot of time. To decrease the time consumption, we calculated all the shortest paths between each pair of road segments based on the Dijkstra algorithm and recorded the data into files. And this preprocessing step can hugely decrease the time consumption of the whole method.

5. Experiment and Result

In this section, we introduce the ways we used to evaluate the proposed method and results of the experiments.

5.1. Experimental Setup. Because the mobile phone data used in our experiment are all anomalous, we cannot directly obtain individuals real commute route. Another issue is that people may not always pass the same route every day. So to testify to the effectiveness of our method in recovering individual's commute route, we adopt two ways of evaluation: path score evaluation and visual evaluation. Path score evaluation grades each path and compares paths based on their score. Visual evaluation draws all the commute information on the map and compares paths visually.

To the best of our knowledge, there exists no model directly generating individual's commute route based on mobile phone data of multiple days. Then we choose the method which is always used in calculating the commute distance [35] as the baseline compared method. The compared method treats the shortest path from home to work based on the real world road network as the approximate commute path. And we compare the paths generated by the two methods through path score evaluation and visual evaluation. Due to the particularity of the experiment, we randomly choose 10 cases for the study.

5.2. Path Score Evaluation. To evaluate the quality of the path, we use number of nearby cell towers to measure the path's authenticity. Intuitively we believe that the most regular path will pass most of cell towers. For each road segment in the path, we calculate the number of nearby cell towers which are within 300 meters from each road segment. Then we can obtain the total number of nearby cell towers for the whole path and we treat the number as the score of the path. For each path $P = \{r_1, r_2, \ldots, r_n\}$, where r_i represents the ith road segment of the path, we calculate the score of the path based on the following formula:

$$score = \sum_{i=1}^{n} sizeOf\left(C_i\right), \quad C_i \in C, \tag{4}$$

where C_i represents the set of all the cell towers within 300 meters from the road segment r_i and C refers to all the cell towers contained in one person's history locations.

We randomly extract 10 people's data and generate the paths by the proposed method and compared method separately. Then we calculate the score of each path. The result is shown in Figure 9; from that we can see that all the paths generated by our method perform better than the compared paths except two samples which have the same score for the two paths.

5.3. Temporality Analysis. We fuse the data generated in multiple days to increase the sampling rate of location points around the commute route. To test the effectiveness of data fusion module, we use the data generated in different number of days to recover the commute route by the proposed model; then we evaluate each path by the path score evaluation. The selected periods include one day, one week, two weeks, three weeks, and four weeks. For the one day's data, we actually choose the day that performs best among all the days. As is shown in Figure 11, the score of recovered path increases as the period getting longer. And we can see that the score based

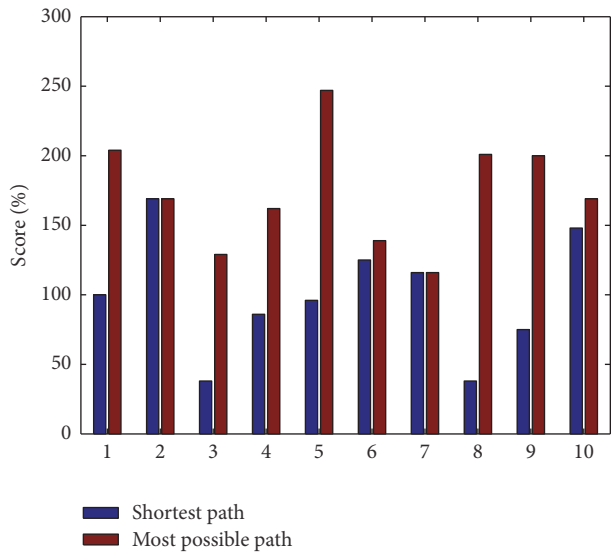
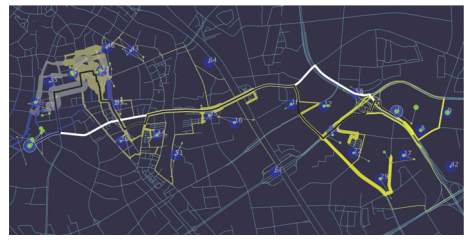


Figure 9: Performance of path score for the two methods.

(a) Commute cell group trajectories of Person A in multiple days



(b) Frequency of each road segment of Person A and his/her shortest path (white line) and most possible path (black line)

(c) Commute cell group trajectories of Person B in multiple days

(d) Frequency of each road segment of Person B and his/her shortest path (white line) and most possible path (black line)

Figure 10: All the commute information of two individuals.

on one week's data has huge improvement than the one based on one day's data and the score tends to stable after the period increases to two weeks.

5.4. Visual Evaluation

5.4.1. Visualization. For better understanding individual's commute route, we draw all the commute information on the map. As mentioned above, people are not always passing the same route every day. For example, people in Beijing are not allowed to drive their own cars during some special days, so they have to take the bus or subway to get to work instead and this can cause the different commute routes for one person. We randomly choose two people's commute information and draw them on the map. We design two kinds of figures for the visualization: one contains the basic information and another contains all the generated routes.

Figures 10(a) and 10(c) show all the cell towers one person passed, all the groups generated by the leader cluster algorithm and commute paths in each day. Each yellow dot represents one specific cell tower and the larger dot means that the corresponding tower recorded much more records for the person. The lines in different color represent the commute route in different days. Finally, the blue circle

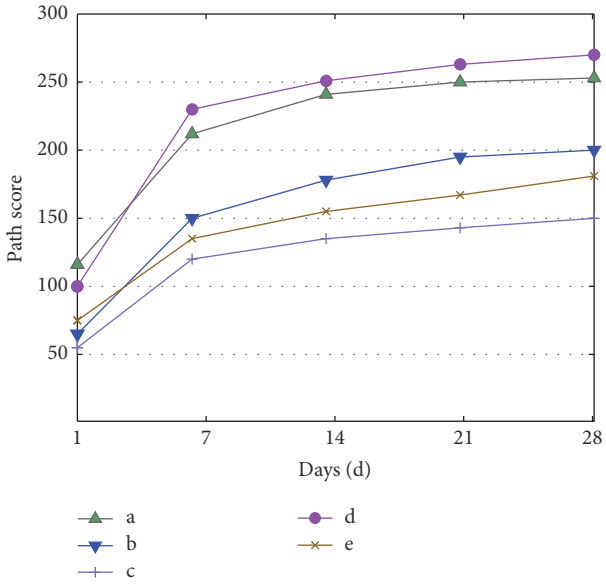


Figure 11: Choose shortest path as commute path.

represents the center of the group and all the members who belong to this group are linked together by the fine yellow lines.

Figures 10(b) and 10(d) include all the road segments one person passed every day, the shortest path from home to work and the path with highest frequency which is generated by our method. The width of each road segment is proportional to its frequency, and the more times one person passes the road segment, the higher frequency the road segment obtains. Besides, we calculate the average time when people passed the road segment and use different colors to represent different period of time. As we can see in Figure 10, the brighter color the road segment has, the earlier time the person will pass the road at. Besides, there are two long paths from home to work: the white one is generated by the baseline method which is exactly the shortest path and the black one is the path generated by our method.

5.4.2. Visual Analysing. Figures 10(a) and 10(c) show all the commute paths of the two people in different days. From these lines we can basically see the main commute path. Besides the main path, there are some paths that are quite different from the main path, which verify supposing that people are not always passing the same route every day. Figures 10(b) and 10(d) draw the road segments with different widths and colors which can better reveal the whole commute information of the people. As we can see from the figure, the black path is surrounded by much more cell towers than the white path and the road segments included in the black path have much higher frequencies than the white path. Then we can conclude that the black path which is generated by our method is much closer to the real commute path than the compared path.

As is shown in Figure 12, there are few situations that people did actually choose the shortest path as the common commute path. That is not common because the shortest path may not be the fastest path when considering the status of the roads.

Figure 12: Choose shortest path as commute path.

6. Conclusion and Discussions

In this paper, we propose a commute route recovering model to recover individual's commute route based on passively generated mobile phone data. The proposed model contains two main modules to deal with different tasks. The data preprocessing module applies leader clustering algorithm to deal with the challenge of low precision of location information. The map matching module calculates the transfer frequency of all related road segments by fusing multidays' commute paths with road network to deal with the challenge of low sampling rate of signal records and multiple overlapping paths. The model generates the path with the highest possibility as the commute path. We adopt two ways to evaluate the result. Experiments show better performance of our model than the compared method.

To the best of our knowledge, our work is the first to explore recovering people's commute route based on the mobile phone data in multiple days. So inevitably the proposed model may have several limitations. For example, the model is sensitive to the quality of the real world road network and that will determine the precision of the generated path to a certain extent. Besides, how to generate the more authentic paths from the transfer matrix in a better way is the aspect we will keep exploring.

Competing Interests

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61572059), the National Natural Science Foundation of China (no. 51408018), the State Key Program of National Natural Science of China

(Grant no. 71531001), the State's Key Project of Research and Development Plan (2016YFC1000307), and the Program of Shenzhen (JCYJ20150624154400509).

References

[1] D. Soper, "Is human mobility tracking a good idea?" *Communications of the ACM*, vol. 55, no. 4, pp. 35–37, 2012.

[2] J. Tang, F. Liu, Y. Wang, and H. Wang, "Uncovering urban human mobility from large scale taxi GPS data," *Physica A: Statistical Mechanics and Its Applications*, vol. 438, pp. 140–153, 2015.

[3] M. Zilske and N. Kai, "A simulation-based approach for constructing all-day travel chains from mobile phone data," *Procedia Computer Science*, vol. 52, no. 1, pp. 468–475, 2015.

[4] C. Chen, L. Bian, and J. Ma, "From traces to trajectories: how well can we guess activity locations from mobile phone traces?" *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 326–337, 2014.

[5] R. Ahas, S. Silm, O. Järv, E. Saluveer, and M. Tiru, "Using mobile positioning data to model locations meaningful to users of mobile phones," *Journal of Urban Technology*, vol. 17, no. 1, pp. 3–27, 2010.

[6] N. E. Williams, T. A. Thomas, M. Dunbar, N. Eagle, and A. Dobra, "Measures of human mobility using mobile phone records enhanced with GIS data," *PLOS ONE*, vol. 10, no. 7, Article ID e0133630, 2015.

[7] J. Blumenstock, G. Cadamuro, and R. On, "Predicting poverty and wealth from mobile phone metadata," *Science*, vol. 350, no. 6264, pp. 1073–1076, 2015.

[8] J.-P. Onnela and A.-L. Barabsi, "Structure and tie strenghs in mobile communication networks," *Proceedings of the National Academy of Sciences*, vol. 104, no. 18, pp. 7332–7336, 2007.

[9] X. Pan, W. Chen, L. Wu, C. Piao, and Z. Hu, "Protecting personalized privacy against sensitivity homogeneity attacks over road networks in mobile services," *Frontiers of Computer Science*, vol. 10, no. 2, pp. 370–386, 2016.

[10] B. C. Csáji, A. Browet, V. A. Traag et al., "Exploring the mobility of mobile phone users," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 6, pp. 1459–1473, 2013.

[11] F. Calabrese, F. C. Pereira, G. D. Lorenzo et al., "The geography of taste: analyzing cell-phone mobility and social events," in *Proceedings of the International Conference on Pervasive Computing*, pp. 22–37, Springer, 2010.

[12] S. Isaacman, R. Becker, R. Cáceres et al., "Identifying important places in peoples lives from cellular network data," in *Pervasive Computing: 9th International Conference, Pervasive 2011, San Francisco, USA, June 12–15, 2011. Proceedings*, vol. 6696 of *Lecture Notes in Computer Science*, pp. 133–151, Springer, Berlin, Germany, 2011.

[13] Y. Zheng, "Methodologies for cross-domain data fusion: an overview," *IEEE Transactions on Big Data*, vol. 1, no. 1, pp. 16–34, 2015.

[14] Y. Zheng, Y. Liu, J. Yuan, and X. Xie, "Urban computing with taxicabs," in *Proceedings of the 13th International Conference on Ubiquitous Computing (UbiComp '11)*, pp. 89–98, ACM, Beijing, China, September 2011.

[15] N. J. Yuan, Y. Zheng, and X. Xie, "Segmentation of urban areas using road networks," Microsoft Technical Report, 2012.

[16] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[17] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Exploiting multi-channels deep convolutional neural networks for multivariate time series classification," *Frontiers of Computer Science*, vol. 10, no. 1, pp. 96–112, 2016.

[18] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML '11)*, pp. 689–696, Bellevue, Wash, USA, July 2011.

[19] Q. Zhang, L. T. Yang, and Z. Chen, "Deep computation model for unsupervised feature learning on big data," *IEEE Transactions on Services Computing*, vol. 9, no. 1, pp. 161–171, 2016.

[20] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[21] Q. Zhang, H. Zhong, L. T. Yang, Z. Chen, and F. Bu, "PPHOCFS: privacy preserving high-order CFS algorithm on the cloud for clustering multimedia data," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 4, article 66, 2016.

[22] J. Xin, Z. Cui, P. Zhao, and T. He, "Active transfer learning of matching query results across multiple sources," *Frontiers of Computer Science*, vol. 9, no. 4, pp. 595–607, 2015.

[23] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with cotraining," in *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT '98)*, pp. 92–100, ACM, Madison, Wis, USA, 2000.

[24] Q. Wang, B. Wang, X. Hao et al., "Face recognition by decision fusion of two-dimensional linear discriminant analysis and local binary pattern," *Frontiers of Computer Science*, vol. 10, no. 6, pp. 1118–1129, 2016.

[25] Y. Zhang, J. Zhang, Z. Pan, and D. Zhang, "Multi-view dimensionality reduction via canonical random correlation analysis," *Frontiers of Computer Science*, vol. 10, no. 5, pp. 856–869, 2016.

[26] Q. Zhang, L. T. Yang, Z. Chen, and F. Xia, "A high-order possibilistic-means algorithm for clustering incomplete multimedia data," *IEEE Systems Journal*, pp. 1–10, 2015.

[27] Q. Zhang and Z. Chen, "A weighted kernel possibilistic c-means algorithm based on cloud computing for clustering big data," *International Journal of Communication Systems*, vol. 27, no. 9, pp. 1378–1391, 2014.

[28] W. Rong, B. Peng, Y. Ouyang, K. Liu, and Z. Xiong, "Collaborative personal profiling for web service ranking and recommendation," *Information Systems Frontiers*, vol. 17, no. 6, pp. 1265–1282, 2015.

[29] J. S. Greenfeld, "Matching GPS observations to locations on a digital map," in *Proceedings of the Transportation Research Board Meeting*, National Research Council (US), Washington, DC, USA, 2002.

[30] S. Brakatsoulas, D. Pfoser, R. Salas et al., "On map-matching vehicle tracking data," in *Proceedings of the 31st International Conference on Very Large Data Bases*, Trondheim, Norway, September 2005, https://www.researchgate.net/publication/221310236_On_Map-Matching_Vehicle_Tracking_Data.

[31] Y. Zheng, "Trajectory data mining: an overview," *ACM Transactions on Intelligent Systems and Technology*, vol. 6, no. 3, article no. 29, 2015.

[32] J. Yuan, Y. Zheng, C. Zhang, X. Xie, and G.-Z. Sun, "An Interactive-Voting based Map Matching algorithm," in *Proceedings of the 11th IEEE International Conference on Mobile Data*

Management (MDM '10), pp. 43–52, IEEE, Kansas City, Mo, USA, May 2010.

[33] A. Thiagarajan, L. Ravindranath, H. Balakrishnan et al., *Accurate, Lowenergy Trajectory Mapping for Mobile Devices*, Networked Systems Design and Implementation, 2011.

[34] Q. Wu, X. Qi, E. Fuller, and C.-Q. Zhang, ""Follow the leader": a centrality guided clustering and its application to social network analysis," *The Scientific World Journal*, vol. 2013, Article ID 368568, 9 pages, 2013.

[35] P. Yang, T. Zhu, X. Wan, and X. Wang, "Identifying significant places using multi-day call detail records," in *Proceedings of the 26th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '14)*, pp. 360–366, IEEE, Limassol, Cyprus, November 2014.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 3254908, 13 pages
https://doi.org/10.1155/2017/3254908

Research Article

A Process Mining Based Service Composition Approach for Mobile Information Systems

Chengxi Huang, Hongming Cai, Yulai Li, Jiawei Du, Fenglin Bu, and Lihong Jiang

School of Software, Shanghai Jiao Tong University, Shanghai, China

Correspondence should be addressed to Hongming Cai; hmcai@sjtu.edu.cn

Received 23 September 2016; Revised 29 November 2016; Accepted 18 December 2016; Published 23 January 2017

Academic Editor: Laurence T. Yang

Due to the growing trend in applying big data and cloud computing technologies in information systems, it is becoming an important issue to handle the connection between large scale of data and the associated business processes in the Internet of Everything (IoE) environment. Service composition as a widely used phase in system development has some limits when the complexity of relationship among data increases. Considering the expanding scale and the variety of devices in mobile information systems, a process mining based service composition approach is proposed in this paper in order to improve the adaptiveness and efficiency of compositions. Firstly, a preprocessing is conducted to extract existing service execution information from server-side logs. Then process mining algorithms are applied to discover the overall event sequence with preprocessed data. After that, a scene-based service composition is applied to aggregate scene information and relocate services of the system. Finally, a case study that applied the work in mobile medical application proves that the approach is practical and valuable in improving service composition adaptiveness and efficiency.

1. Introduction

Along with the rapid advancements in big data and cloud computing technologies, connection of everything is emphasized in many information systems. Thanks to the achievements of devices, infrastructure, and applications in mobile computing [1, 2], systems become more powerful and intelligent with the support of connection among devices, people, and business processes. Particularly, according to the recent research [3], mobile technology development has resulted in the creation of up to 1450,000 applications for smart phones in the last few years. More and more information systems rely on service-oriented processes in order to fit the continually changing business environment and to align business strategies with IT systems [4]. With strong interaction with people and social environments, these systems have a great impact in many areas such as health care [5, 6], exploiting indoor location [7], and other scenarios. As a result, it is becoming more and more valuable to deal with the connection among devices and interaction among people especially in the environment of the Internet of Everything (IoE).

Due to the flexible and scalable characteristics of service-oriented computing, more and more systems use web services composition to deal with the complexity of multisource data in mobile information systems. Business processes and associated services become the most significant supports for the connection of everything. They make functions and devices work as expected in well-organized systems. Achieving adaptiveness in process-based service composition is the key to improve efficiency and adaptiveness of mobile systems.

However, as both the scale and the variety of devices are expanding, the complexity of service implementation is increasing. To sum up, challenges exist in keeping the system process adaptive to the changing environment as the following points:

(1) Process execution environment is changing: in the environment of IoE, as users, devices, and services are widely distributed, the execution of the process may be affected by changing device rules, connection situations, and event users' habits. As more complex rules are introduced with the devices, static processes always lack the consideration of execution environment, and they cannot handle the changing environment efficiently. For instance, in mobile systems, different versions of applications are used at the same

time, which will make the processes in the server side suffer from errors if they cannot handle the changing orders of events.

(2) The complexity of relationship in events and services is increasing: since types of devices are increasing, the relationship in events and services is getting more complicated. Current process-based service composition is not flexible enough to support the complex situations. As a result, approaches designed for application execution are usually incomplete and lacking necessary business consideration. For example, in a smart house application, when new devices like new models of air conditioners are introduced, new events and new connections will be introduced and the controlling process should be fixed accordingly in order to keep the devices and services work correctly.

In our previous work [8], the service composition based on process mining approach has been applied to a logistics cloud service platform which supports the users from different companies to customize their functional services. In the example case about the waybill transportation process, a suitable waybill-related composite service is generalized to connect the information sensing devices like radio frequency identification (RFID), infrared sensors, global positioning system (GPS), and laser scanner. And it is proved that service composition based on process mining is suitable for the situation with indefinite requirements and without high performance demand of the result composite service. Considering the expanding scale and the variety of devices in mobile information systems, a process mining based service composition approach is proposed based on our previous work in this paper in order to improve the adaptiveness and efficiency of compositions.

Generally speaking, the main contributions in this paper can be summarized as follows:

(i) Firstly, to solve the problems above, process mining based service composition is proposed to produce adaptive service composition according to real execution information. A three-step framework is presented to cover the whole life cycle of service composition based on process mining.

(ii) Secondly, according to the framework, a set of models is put forward to support the holistic service composition approach which covers both the practical business and the execution effectiveness.

(iii) Then, to apply request-based logs in event-based process mining, a preprocessing algorithm is presented to transfer request-based logs to event-trace-based models so that the execution data can be used in process mining.

(iv) Last but not least, a scene-based service composition algorithm is presented in order to transfer the process mining results to service composition models which can be further used in service generation.

The remaining parts of the paper are organized as follows: in Section 2, an overall description of the proposed approach is provided. After that, the formal analysis and algorithms in context-based service matching is described in Section 3. And then a case study is presented to validate the method in this approach in Section 4, followed by a brief discussion and comparison of the related works in Section 5. Finally, conclusions and future works are given in Section 6.

2. Overview of Process Mining Based Service Composition

In the environment of IoE, large amounts of event-based devices are involved in information systems. Each of them has individual rules due to the differences in types of devices, users, and execution context. In certain situation, they invoke a set of services to provide and retrieve data as well as execute special functions. Behind the devices, the server-side business processes which represent the sequences of service execution and service composition take the role to ensure the functional correctness of the whole system in either explicit or implicit way.

Process mining [11] is a process management technique that extracts information from event logs recorded by an information system to discover, analyze, and enhance process models. Service discovery mining is one of the most potential applications of the state-of-the-art process mining technologies [12]. It includes discovering service behavior, checking conformance of service, and extending service model based on event data. The processes discovered by process mining can provide the best practices during the execution period. The discovered processes with frequently used services can be regarded as composite web service patterns to help the developing of service composition. To improve the fitness of event rules applied in widely spread devices and the business process maintenance in information systems, three concerns are involved, namely, the execution log from IoE environment, control flow analysis for server side, and the service composition.

In order to cover the life cycle, three phases in process mining based service composition are proposed in this paper, as shown in Figure 1.

First, the approach preprocesses the execution data from current system by extracting the service logs and transforms them into valid traces. Then we leverage process mining algorithms to mine the control flow with the result of the previous step. After that, a metamodel is designed to connect the information of execution environment existing business rules, and service deploy model is generated after relocation the service mapping. The description of the steps is as follows:

(i) The first phase is to preprocess device request services:

(a) input: service invocation records, event rules;
(b) output: Trace Model.

(ii) The second phase is to mine process from event traces:

(a) input: Trace Model;
(b) output: Control Flow Model.

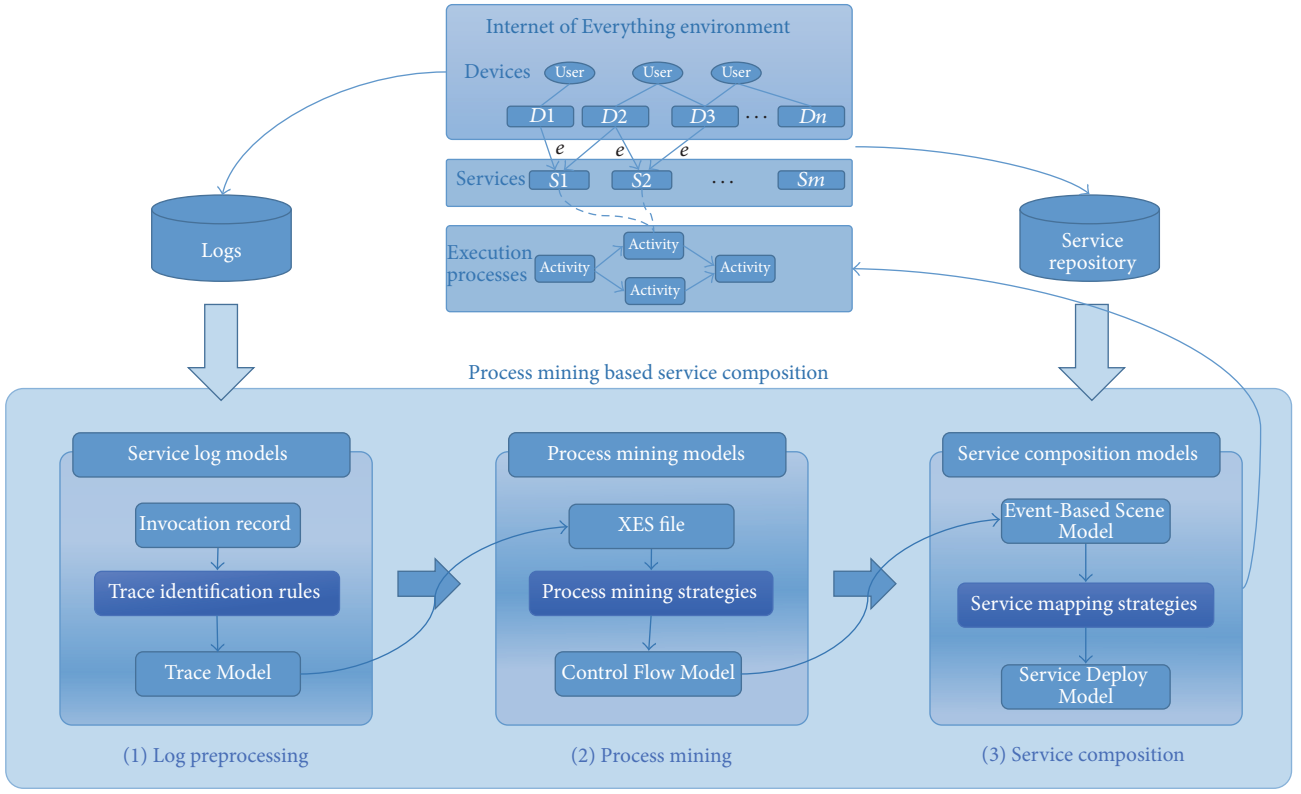


Figure 1: Framework of process mining based service composition, which is divided into three gradual phases: (1) log preprocessing, (2) process mining, and (3) service composition.

(iii) The third phase is to assist service composition with the produced control flow:

(a) input: Control Flow Model, service list;

(b) output: Service Deploy Model.

With execution information retrieved by preprocessing log data, the approach produces a service deploy model for constructing service compositions that is more accurate to the requirement in IoE. Afterwards, new logs will be recorded during the execution of the composite service; therefore the whole life cycle of the service composition procedure becomes a closed loop.

3. Process Mining Based Service Composition

In the following part, the framework mentioned above will be refined to introduce its specifics.

3.1. Models for Process Mining Based Service Composition. A set of models are defined in order to cover the life cycle of process mining based service composition in the three phases of the approach. Figure 2 shows three sets of models and their relationships involved in our approach, including Service Log Models, Process Mining Models, and Service Composition Models.

3.1.1. Service Log Models. Service log models are the set of models that cover preprocessing procedure. The included models are Invocation Log Model, Service Event Model, and Trace Model as the following definitions.

Definition 1. InvocationLogModel (ILM) represents the invocation records that devices executed as event requests. It is a list of service invocation records containing information of devices, users, services and the execution timestamp. The definition of ILM is as equation (1)–(4).

$$ILM \longleftarrow \{Device, Service, User, Timestamp\}, \quad (1)$$

$$Service \longleftarrow \{Description, RequestURL, Action\}, \quad (2)$$

$$Devices \longleftarrow \{DeviceID, DeveiceType, OSType\}, \quad (3)$$

$$User \longleftarrow \{UserID, UserName, RoleSet, UserProperties\}. \quad (4)$$

Definition 2. EventDictionaryModel (EDM) represents the dictionary of the mapping rules between events and the execution services, as shown in (5). The event is defined as in (6), and the service shares the same definition as that in (2):

$$EDM \longleftarrow \{Event, Service\}, \quad (5)$$

$$Event \longleftarrow \{EventName, EventCategory\}. \quad (6)$$

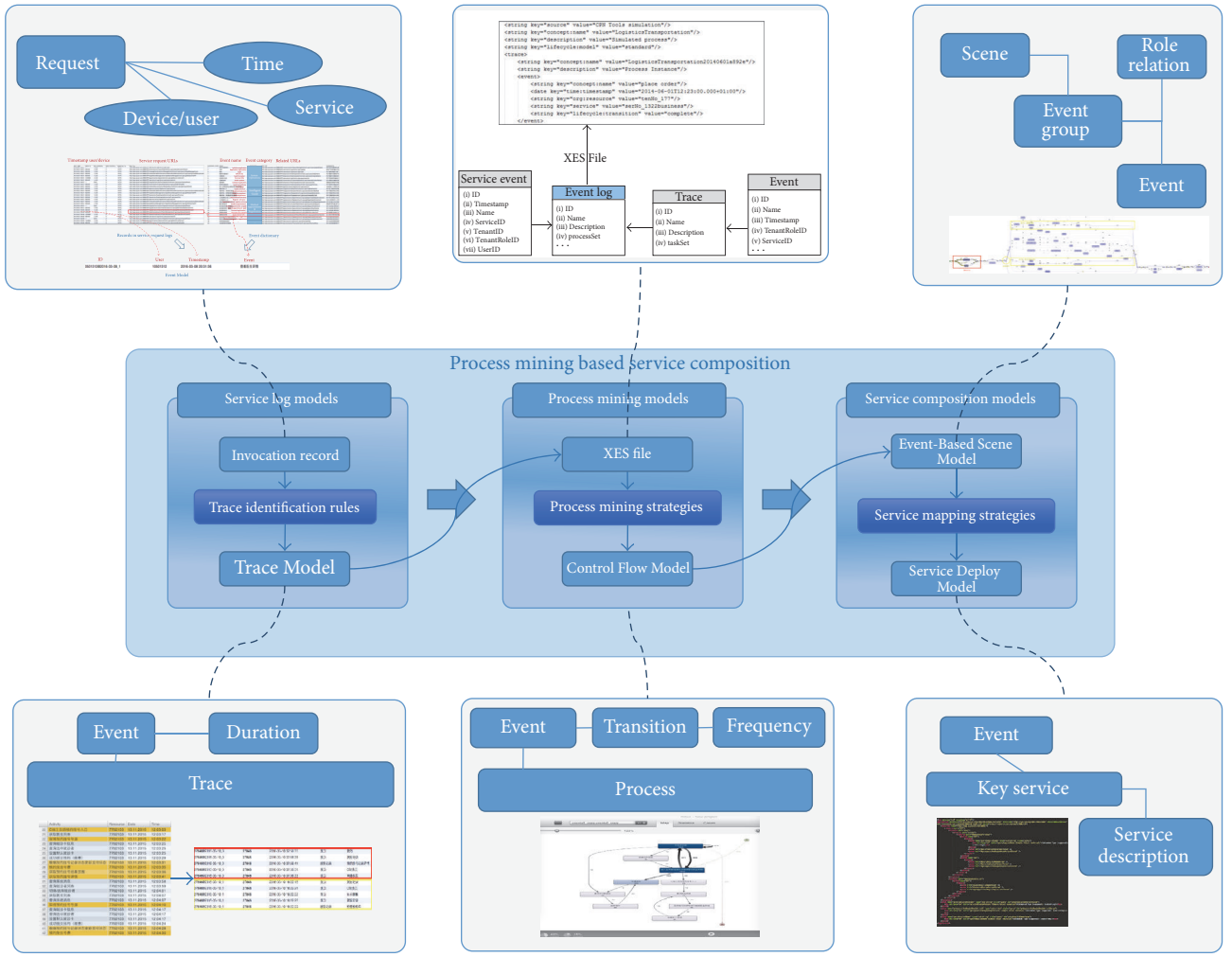


Figure 2: Process mining based service composition models.

Definition 3. EventModel (EM) keeps the operation information from users, including *User, Event,* and *Timestamps*, as in the following equation:

$$EM \longleftarrow \{User, Event, Timestamps\}. \quad (7)$$

Definition 4. TraceModel (TM) contains a group of traces that represent a sequence of continual operation events, including a set of event models and the time duration information, as in the following equation:

$$TM \longleftarrow \{List\{EM\}, User, StartTime, EndTime\}. \quad (8)$$

3.1.2. Process Mining Models. The process mining model restores information for process mining.

The Extensible Event Stream (XES) can be regarded as unification data form between trace models and standard process mining input. The input format of this phase is XES which is a process instance that has integrated multiple Service Events. It contains multiple processes, which are called trace in XES standards, and every trace is related to a trace model that contains multiple events.

Definition 5. ProcessModel (PM) is the output of process mining. Business process is defined as a process that contains events and the control flow between them which is presented as event and transition. And a set of frequency representing the execution frequency of each event is also included for further analysis, as in the following equation:

$$PM \longleftarrow \{Event, Transition, Frequency\}. \quad (9)$$

3.1.3. Service Composition Models. The service composition models restore information from process model, event-role relation, and event service relation. Process model is the process discovered through process mining. Event-role relation includes relations between service events and roles. And Key Service Model is the mapping between services and scene-based events.

Definition 6. EventServiceSceneModel (ESRM) represents the scene based on event analysis:

$$ESRM \longleftarrow \{SceneDescription, ServiceSet, EventSet, RoleSet, KSM\}. \quad (10)$$

Input:
IvoncationLogModel: *ILM*,
EventDictionaryModel: *EDM*,
ValidateUser
TimeDuration
Output:
TraceModelTM
(1) *FilteredLog* ← *ILM*
(2) *TM* ← ∅
(3) *FilteredLog*.remove r if not $User(r) \in ValidateUser$
(4) *FilteredLog*.remove r if not $Time(r) \in TimeDuration$
(5) *FilteredLog*.remove r if $r' == r$
(6) *FilteredLog.sortByTime*()
(7) *trace* ← {*FilteredLog.first*().*event*()}
(8) **while** *FilteredLog.hasNext*() **do**
(9) r ← *FilteredLog.next*().*event*()
(10) **if** r in a short time **then**
(11) trace.add(r)
(12) **else**
(13) TM.add(trace)
(14) *Trace* ← {r}
(15) **end if**
(16) **end while**
(17) **return** *TraceModel*

Algorithm 1: Preprocessing—preprocessing logs for process mining by steps of removing invalid records, eliminating similar request in a short time, picking the successful request and deleting others, connecting service with events, and dividing the events into traces, according to time duration.

Definition 7. KeyServiceModel (KSM) represents the mapping between events and most suited services:

$$KSM \longleftarrow Set\{Service, Event\}. \tag{11}$$

3.2. Execution Log Processing. The log data in IoE is getting more complex with increasing amount of connections, leading to larger scale of events and services. As a result the service logs are not suitable for process mining due to noises and unclear boundaries. Therefore, in the first phase of our method, we extract the execution data from service logs, remove the noise data, and generate traces in trace model.

The preprocessing algorithm is shown as Algorithm 1. Consider the record size of initial logs as data size n. The data cleaning part (line (1) to line (5)) takes a time complexity of $O(n)$, for we only have to travel the data once and remove dirty data by $O(1)$ determinations. And the sorting part (line (6)) is a classic sorting problem which can be optimized to finish in $O(n \log n)$. Finally, the connecting part (the while loop) takes the time complexity of $O(n)$. Because we go through the clean logs (less than n) again and the creating of trace is an $O(1)$ operation, the overall complexity of the algorithm is $O(n \log n)$. As we can see, the preprocessing procedure uses most time in sorting the event records. If the records are already sorted in the initial logs, this algorithm can have a time complexity of $O(n)$. As to space requirement, the cleaning part can be done in place. The sorting part and connecting part each take $O(n)$ space. Because the data size n can be controlled by separating logs by different time periods, this step can be done distributively in acceptable time. Therefore the preprocessing step will not take too much time regarding large scale of logs.

3.2.1. Preprocessing Noise Data. In preprocessing phases, first of all, service invocation logs are used as input of preprocessing step. The original logs keep recordings of service invocation information. Logs contain information for process execution and bridge the gap between service composition and service deployment. However, the logs cannot be used as input of process mining directly as a result of different viewpoints of data organization and different structures of data storage. Therefore, before doing process mining, it is necessary to remove the outdated and incorrect data in logs to extract the required information.

First of all, we manually decide valid users, valid time, and max transaction duration, which means to define *Set*{*ValidateUser*} and *Set*{*TimeDuration*}. Then we remove the invalid records according to the valid configuration. After that, we eliminate the duplicate records that are produced due to connection errors in network.

3.2.2. Generating Event Model. The next step is to transform the records into the event models with the assistance of event dictionary. As mentioned above, the original service invocation logs are restored in the form of *ILM*. And the process mining are based on event data like *EM*. So we transform the ILM into EM by mapping the attribute of *ILM.service.URL* and *EDM.service.URL*, which is presented as *event*() in the algorithm.

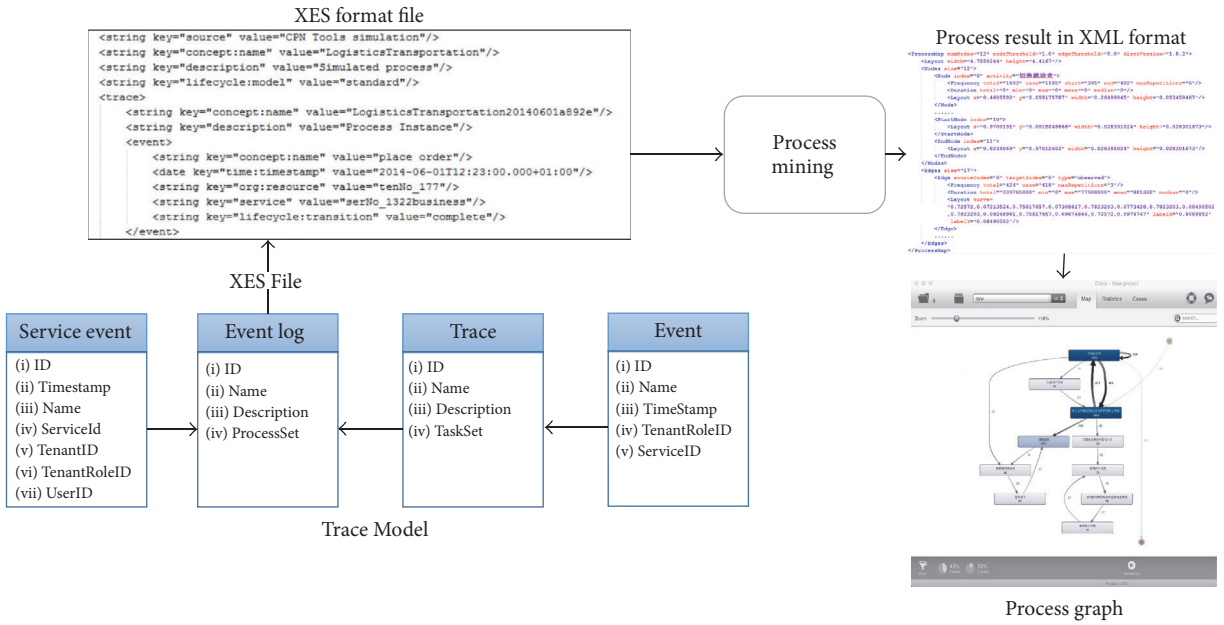


Figure 3: Execution of process mining.

3.2.3. Generating Trace Model. The last step of preprocessing is to reorganize the event models into trace models. Other than the Iterative Expectation-Maximization Procedure method introduced in [13], which takes too much time when confronting large amount of logs, we use the dividing strategy based on time duration separation. First, we group the event models by the attribute of user. That is, for each user, we have a group of (event, timestamps) pairs. By sorting the events on time, the group of events contains sequences of events. Then we separate them into different traces according to the time duration.

3.3. Process Mining. Process mining is a technique that extracts information from event logs recorded by an information system to discover, analyze, and enhance process models. As in Figure 3, the event logs are from the executing network of devices.

3.3.1. Transforming Trace Model to XES. Processing event logs is to convert the information for process mining we got from log processing into the input criterion required by the process mining tool (like ProM [14] and Disco [15]), which requires XES (Extensible Event Stream) as input format. XES file is a process instance that has integrated multiple service events. It contains multiple processes, which are called trace in XES standards, and every trace contains multiple events, as in the left part of Figure 3.

3.3.2. Executing Process Mining. In the part of process mining, the fuzzy mining algorithm [16] is selected. In the case of our implementation, we choose the fuzzy miner module of tool Disco. The miner is based on the significance and correlation of events to produce adaptable process models, as in the right part of Figure 3.

3.4. Scenario-Based Service Composition. After the steps mentioned above, the process model is produced from device-to-service invocation log. The next step is to adjust the process by execution frequency of events and relocate the services to the process. We provide the procedure as Algorithm 2.

Consider the total event size as data size n. Removing less important nodes (line (1) to (5)) takes $O(n)$, because we only have to calculate the result of $\sum_{e_i \in E} F(e_i)$ once. And in the event grouping and scene generalization part (line (7) to line (16)), calculating all the $\text{sim}(e_i, e_j)$ takes $O(n^2)$. And add/remove operation can be done in $O(1)$. Since the while loop iterates at most n times, the worst complexity of the algorithm is $O(n^3)$. As we can see, the most time taken is in generating Composition Model. The iteration time is dependent on specific data. Comparing to other composition approaches, the scenario generation takes extra time to simplify the processes. Since the event size will not be very large in systems, the time consumed is considered acceptable.

3.4.1. Scene-Based Event Analysis. As a process mining result, a mined process is presented as a directed graph with nodes and edges. By analyzing the source and target in process model, we could get the sequence of events in a process graph. In the graph, nodes represent events and edges indicate the transitions of events. Each edge has a weight representing the frequency of transitions.

To simplify the graph, insignificant nodes and edges will be removed. Frequency of an event e is noted as $F(e)$. Then the importance of the event $I(e)$ is defined as

$$I(e) = \frac{F(e)}{\sum_{e_i \in E} F(e_i)}. \tag{12}$$

Input:
$PM = \{V, E, F\}$
Output:
CompositionModel
(1) **for all** $e_i \in E$ **do**
(2) **if** $I(e) < IO(e)$ **then**
(3) $E.remove(e)$
(4) **end if**
(5) **end for**
(6) Composition Model = PM
(7) **while** Last iteration change Composition Model **do**
(8) **for all** $e_i \in E$ **do**
(9) **for all** $e_j \in e_i.to$ **do**
(10) **if** $\text{sim}(e_i, e_j) < threhold$ **then**
(11) $CompositionModel.remove(e_i, e_j)$
(12) $CompositionModel.add(Scene(e_i, e_j))$
(13) **end if**
(14) **end for**
(15) **end for**
(16) **end while**
(17) **return** Composition Model

Algorithm 2: Scene-based service composition.

Thus $I(e)$ is the ratio of its frequency $F(e)$ and the sum of all the event frequencies. The events with much low frequency can be removed from the graph.

And for the edges, we note sum of all the input transition frequencies as $ID(e)$ and sum of all the output transition frequencies as $OD(e)$:

$$IO(e) = \frac{ID(e)}{OD(e)}. \tag{13}$$

The smallest $IO(e)$ is the start node of the process, and the largest is the end node.

For a transition t, and its source event $e = t.sourceEvent$, the importance of the transition $I(t)$ is shown as follows:

$$I(t) = \frac{F(t)}{F(e)}. \tag{14}$$

If $I(t)$ is much lower than normal, the transition hardly happens according to existing logs. So it can be removed:

$$\text{sim}\left(e_i, e_j\right) = \frac{F\left(t_{ij}\right)}{DO\left(e_i\right) + IO\left(e_j\right)}. \tag{15}$$

For the nodes with similarity close to 1, they are normally executed as a patterned sequence. In other words, e_i, e_j are usually executed at the similar situations. We can group (e_i, e_j) as a scene. And this procedure is repeated literately.

3.4.2. Determine Key Services. In this part, services are marked with priorities in order to pick the most suitable service for each event. In the service repository, similar services are existing. However, these services have different influence in a particular process environment. It is necessary to pick out the most suitable services.

After process mining, two factors can be introduced in service selection: relevance of service-to-event and relevance of service-to-scene. For each event, each service has a priority. The same event may not invoke the fixed service every time, and one service may also be provided to multiple events, so we need a method to choose suitable services, that is, the strategy we use to extract Key Service from all the invoked services (in service repository). We calculate the weight of the service for the event to measure its criticality in service mapping. $F(e, s)$ represents the number of execution time from service e. The outdated data is filtered, so $F(e, s)$ can be used to calculate the importance of service s to event e:

$$\begin{aligned}\text{priority}\left(e_i, s_i\right) = u \cdot \frac{F\left(e_i, s_i\right)}{F\left(e_i\right)} + v \\ \cdot \frac{\sum_{e \in \text{scene}(e_i)} F\left(e, s_j\right)}{\sum_{e \in \text{scene}(e_i)} F(e)}.\end{aligned} \tag{16}$$

With the priority, each event can be related to most usually used services, which means $KSM \leftarrow Set\{Service, Event\}$ can be generated. And the combination of Composition Model and KSM Model becomes the Service Deployment Model.

4. Evaluation

4.1. Case Study: An Application in Mobile Medical System. In this section a case study will be presented to demonstrate the approach.

One of the most potential usages of connecting everything is the application of IoE in medical processes.

For case study, a mobile medical system with large numbers of smart devices (mostly smart phones) in China

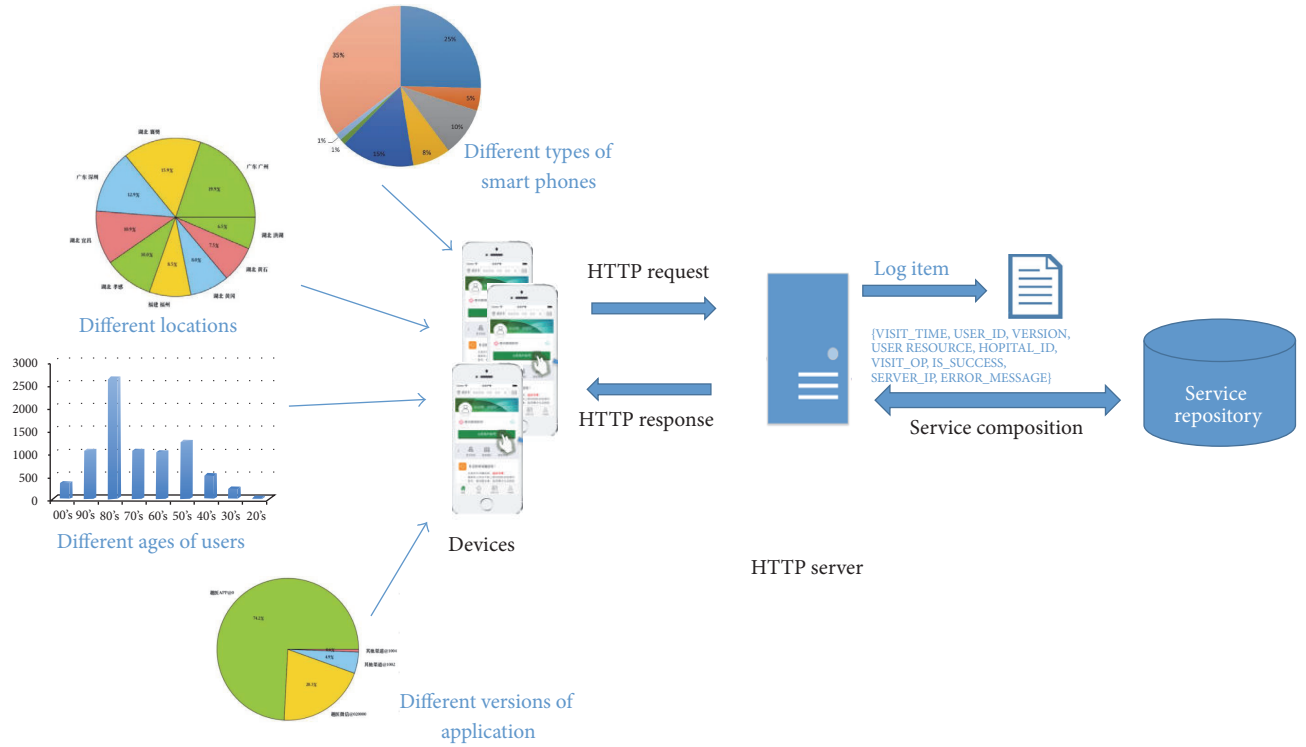


Figure 4: Mobile medical system in IoE.

is used in this evaluation (as in Figure 4). In particular, a registration process is demonstrated in the following part.

As the mobile medical system is getting popular, it is widely used in many provinces over the whole country. The connection network of people, devices, and medical organizations is getting larger recently. With larger scale of usage, the system faces difficulties in optimization of services. The devices have different operation systems and application versions. Due to the variability of operation systems, application versions, and geological locations, the behavior of usage cannot be unified. Unpredictable service usage leads to difficulty in optimization of services. It is inconvenient for updating both mobile applications and server-side systems.

4.2. Preprocessing the Logs. For the case study, five months of logs from the http server of the system is used. The selected logs are from May 2015 to April 2016. Each record includes *visit_time*, *user_id*, *app_version*, *hospital_id* and *visit_op*. The initial log is shown in the left part of Figure 5.

In this log, each record represents a service request. Typical noise of the data includes duplicate operations, invalid operations, and unclear transaction boundary. First, data cleaning is applied to the initial logs. Then, we execute *visit_optobusiness_codemapping*. And the structure of event dictionary is shown in the right part of Figure 5. After mapping service request URL with events, each record is transformed into event model as the bottom part in Figure 5.

To identify traces, the following rules are applied: to ensure over 75% traces are correctly identified, operations that take less than 30 min and 36 seconds are regarded as the same trace. And the result of Trace Model is shown in Figure 6.

4.3. Process Mining. In the process mining phase, the first step is to transfer Trace Model into standard process mining input, that is, to generate XES file with the above method. In the case study, the log is transferred into the log. After preprocessing, we transfer the trace models into XES format, as in Figure 7(a). Disco is chosen to be our process mining platform where the XES can be used directly as standard process mining input. After selecting filters (as Figure 7(b)), we choose the fuzzy miner as the process mining strategy. The tool is used to analyze the interaction records among the business activities in the processes and through mining and reasoning to get the process model. After process discovery, the process model (as in Figure 7(c)) is stored in the form of the XML file (as in Figure 7(d)).

4.4. Scene-Based Composition. Then we combine the service set with the event set. The service is combined with the event according to the corresponding event ID. The similar phase is done to the role set as well. Figure 8 shows the optimization of control flow in this case, which includes start node identification, similar event composition, and less significant event reduction.

Through service selection, a set of key services will be generated. After we import the data of process mining phase to service composition phase, the Service Deployment Model can be generated. And with template technologies, we can generate the service descriptions for service compositions

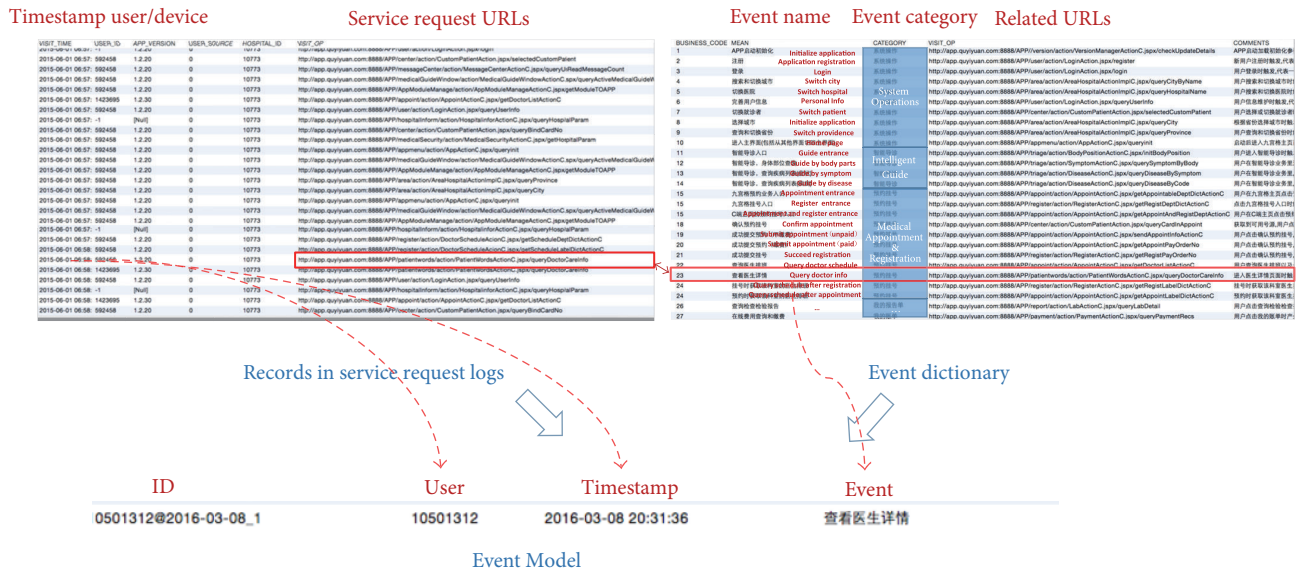


Figure 5: Mapping service request with event dictionary.

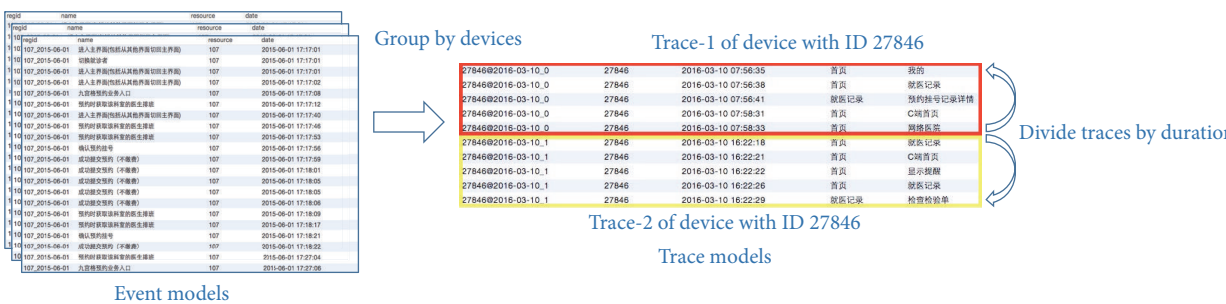


Figure 6: Separating traces from logs.

of scenes. Figure 9 shows examples of result of key services mapping and service generation. In Figure 10(a), the key services are mapped to the events ($u = 0.9, v = 0.1$ for priority calculation). And Figure 10(b) shows one of the examples of generated WSDL descriptions for composited service.

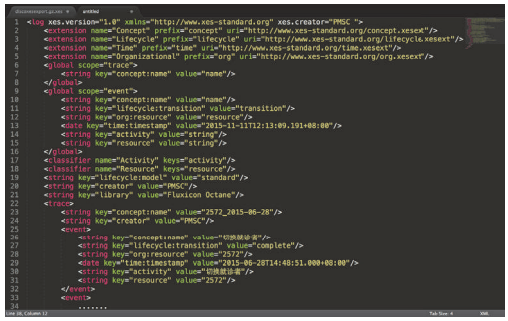
Then the composite service is registered in the service library and enters the service deployment phase. After long-term running, the execution of this service will leave behind service logs which can be used for the new process mining phase of the next generation.

4.5. Result and Discussion. After applying our work to the mobile medical system, the registration process of the system is improved considering two criteria.

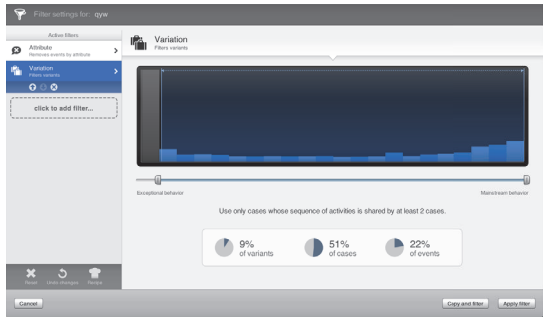
First of all, the simplicity of the new process is improved after we composite the services that invoked as a pattern. Secondly, as services are composited for certain scene, the rules defined in devices can be simplified. And with the discovery of composition, further optimization can be implemented to redeploy the services so that services in the same scene can be physically deployed in the same server to reach a better performance.

We recollect the execution logs after adjustment of event rules to the new service compositions. To evaluate the performance, we compare two log data, one from the week right before redeploying the service composition and the other from the week right after applying our method (see Table 1 and corresponding Figure 10). It is assumed that, in the continuous two weeks, the user behavior and the operation of the application should not change much. As we can see in the result, after reduplicate request and meaningless events are removed, the total amount of the events is reduced owing to the simplification of the process. To complete the same functional requirement, the events of each case are greatly reduced. And the relative percentage of event that may be caused by users' hesitation like "Select City" and "Switch Province" is reduced. Thus the execution of the process is improved by efficiency.

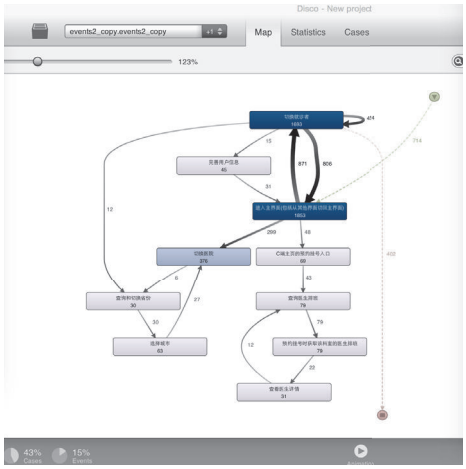
As to privacy issues, first of all, the input of our approach is system log that contains service requests. They do not contain sensitive data such as credit accounts. Our method just uses the necessary data that is usually used for system maintaining. And after process mining, the mining result is a summary of all the behaviors rather than an operation



(a) Snippet of generated XES file



(b) Configuration fuzzy mining



(c) Output of process mining

(d) Output in XML format

Figure 7: Execution of process mining.

Table 1: Comparison of event frequencies before and after our optimization.

Event	Original frequency	Original frequency rate	Improved frequency	Improved frequency rate
Main Entrance	7,777	35.21%	1,853	42.91%
Switch Patient	4,473	20.25%	1,693	39.21%
Check Doctor Schedule	1,825	8.26%	79	1.83%
Query Doctor Info	1,713	7.76%	31	0.72%
Switch Hospital	1,474	6.67%	376	8.71%
Select City	1,114	5.04%	63	1.46%
Appointment Entrance	725	3.28%	69	1.6%
User Profile	621	2.81%	45	1.04%
Switch Province	540	2.44%	30	0.69%

sequence of individual person. So our service composition is based on the summarized result of a group.

5. Related Work

The existing approaches that perform service discovery and service composition will be discussed in this section.

For service selection solutions, in [9], a service selection technique is proposed to select the best potential candidate service from a set of functionally equivalent ones. The approach in [17] takes several aspects such as QoS, user preference, and the service relationship into consideration. And the work [18] proposes an effective approach to extract events and their internal links from large-scale data with predefined event schema.

As to context-aware dynamic service composition approach and AI planning techniques in addition, [10, 19, 20] use models at runtime to guide the dynamic evolution of context-aware web service compositions to cope with unexpected situations. Reference [21] proposes a service granularity space for multitenant service composition, which provides a semantic basis for multitenant service composition. In [22], a methodology based on process mining is proposed to do business process analysis in health care environments to

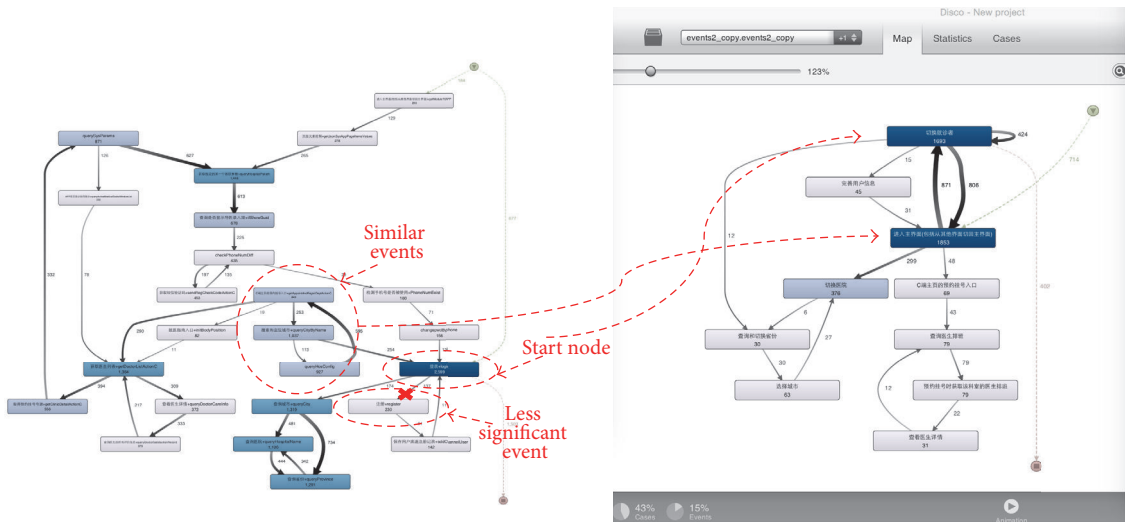
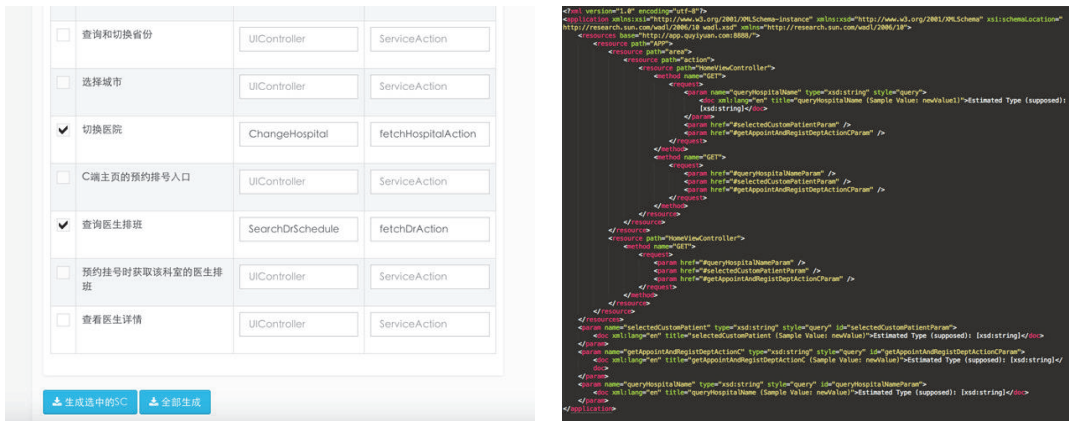


Figure 8: Scene-based control flow optimization.



(a) Service mapping

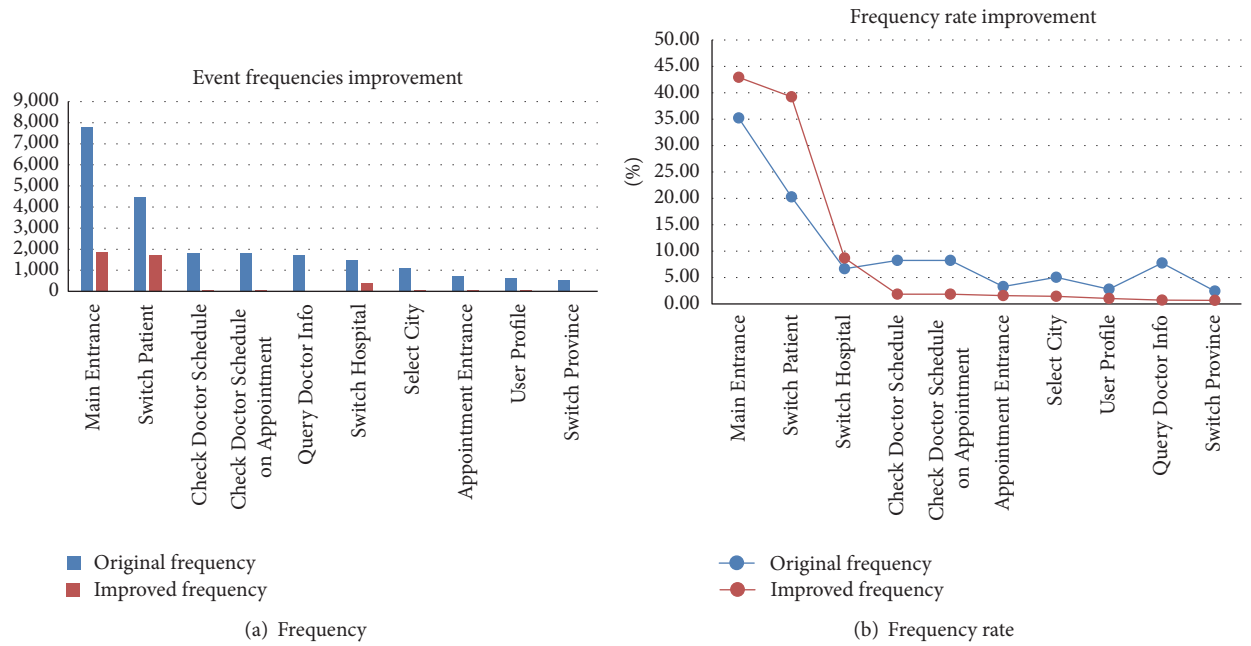
(b) Service generation

Figure 9: Mapping and deploy services.

identify regular behavior, process variants, and exceptional medical cases.

For optimizing the existing service approaches, there are few approaches about service composition in the area of service mining, such as service composition analysis and optimization. The following works are devoted to optimizing the existing service composition based on mining patterns from existing data. A mining algorithm based on statistical techniques to discover composite web service patterns from execution logs is proposed by [23] to better understand, control, and eventually redesign the composite services while [24] proposed an approach to generate service composition pattern for cloud migration from a set of service composition solutions by a graph similarity analysis approach. In [25], an event-based monitoring approach for service composition infrastructures is presented to provide a holistic monitoring approach by leveraging Complex Event Processing techniques. In summary, the works [23–25] use data mining instead of process mining.

Our work proposed a service composition approach based on process mining, which is aimed at improving the adaptiveness and efficiency of compositions considering the expanding scale and the variety of devices in mobile information systems. In terms of the main objectives of these three approaches, our service composition approach is based on process mining and can select services according to the result of the process mining while the other approaches either focus on performance or on context environment. We compare our work with the recent service composition approaches in service composition research area, that is, QoS-based service composition approach [9] and context-aware dynamic service composition approach [10] in Table 2. Although it is hard to execute the data with existing approaches, our approach is more suitable in some cases. Our approach



(a) Frequency

(b) Frequency rate

Figure 10: Comparison of event frequency before and after our optimization.

Table 2: Comparison to existing approaches.

Features	Our approach	QoS-based [9]	Context-aware [10]
Main objective	Improve processes efficiency	Improve performance	Improve adaptabilities
Service selection criteria	Execution info from process mining	QoS goals from SaaS providers	Context collected from equipment
Requirement accuracy	High	Low	High
Nonfunctionality	Medium	High	Low
Adaptability accuracy	High	Low	High
Performance accuracy	Improved	Improved	Medium
Nonfunctionality	Medium	High	Low
Time cost	Lower	Lower	High
Flexibility	Medium	Medium	High
Optimization support	Yes	No	Yes

has advantages that other approaches do not have. Firstly, our work can handle the comprehensiveness from business rules. Rather than focusing on execution time selection as in work [9], service invoking pattern discovery is also considered in our work. As a result, service execution relation can be optimized rather than optimized single request time. Secondly, rather than taking information from equipment context in [10], our method is based on server-side data. Though our offline computing is not as flexible as dynamic perdition, our method can handle a system that different versions of devices rule execute together.

In conclusion, our service composition approach based on process mining is outstanding in comprehensiveness with acceptable time cost and flexibility. However there is currently no standard benchmark to evaluate the performance of each work, due to the different focus of area. It can be concluded that our method can improve both the adaptiveness of functional requirement and the efficiency of process executing. So, it is more suitable than other approaches when there are different types for devices that use services in a different way.

6. Conclusions

In the area of the Internet of Everything, service composition is widely used for the development of applications. In this paper, in order to improve both execution effectiveness and comprehensiveness of existing service compositions, we propose a service composition approach based on process mining, considering both the practical business and the execution information in environment with large amount of connection between devices and users. It is shown that our approach can improve the adaptiveness of process by combining the execution information with service composition. And the efficiency of compositions can be further optimized by redeploying the services in the same scene on the same physical server, which is planned as our further work.

Competing Interests

The authors declare that there are no competing interests.

Acknowledgments

The authors would like to acknowledge the support provided by the National Natural Science Foundation of China under nos. 71171132 and 61373030.

References

[1] H. Hoehle and V. Venkatesh, "Mobile application usability: conceptualization and instrument development," *MIS Quarterly*, vol. 39, no. 2, pp. 435–472, 2015.

[2] F. F. Ntawanga, A. P. Calitz, and L. Barnard, "A context-aware model to improve usability of information display on smartphone apps for emerging users," *The African Journal of Information Systems*, vol. 7, no. 4, p. 3, 2015.

[3] V. V. S. M. Chintapalli, W. Tao, Z. Meng, K. Zhang, J. Kong, and Y. Ge, "A comparative study of spreadsheet applications on mobile devices," *Mobile Information Systems*, vol. 2016, Article ID 9816152, 10 pages, 2016.

[4] H. Cai, C. Xie, L. Jiang, L. Fang, and C. Huang, "An ontology-based semantic configuration approach to constructing Data as a Service for enterprises," *Enterprise Information Systems*, vol. 10, no. 3, pp. 325–348, 2016.

[5] B. Xu, L. D. Xu, H. Cai, C. Xie, J. Hu, and F. Bu, "Ubiquitous data accessing method in iot-based information system for emergency medical services," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1578–1586, 2014.

[6] C.-W. Shen, C.-H. Hsu, C.-C. Chou, and T.-C. Tsai, "Toward a nationwide mobile-based public healthcare service system with wireless sensor networks," *Mobile Information Systems*, vol. 2016, Article ID 1287507, 11 pages, 2016.

[7] H.-Y. Noh, J.-H. Lee, S.-W. Oh, K.-S. Hwang, and S.-B. Cho, "Exploiting indoor location and mobile information for context-awareness service," *Information Processing & Management*, vol. 48, no. 1, pp. 1–12, 2012.

[8] Y. Li, H. Cai, C. Huang, and F. Bu, "Leveraging process mining on service events towards service composition," in *Advances in Services Computing—9th Asia-Pacific Services Computing Conference (APSCC '15)*, pp. 195–209, 2015.

[9] T. Ahmed and A. Srivastava, "Minimizing waiting time for service composition: a frictional approach," in *Proceedings of the IEEE 20th International Conference on Web Services (ICWS '13)*, pp. 268–275, IEEE, Santa Clara, Calif, USA, July 2013.

[10] G. H. Alférez and V. Pelechano, "Facing uncertainty in web service compositions," in *Proceedings of the IEEE 20th International Conference on Web Services (ICWS '13)*, pp. 219–226, IEEE, Santa Clara, Calif, USA, July 2013.

[11] W. Van Der Aalst, A. Adriansyah, A. K. A. De Medeiros et al., "Process mining manifesto," in *Proceedings of the International Conference on Business Process Management*, pp. 169–194, Springer, 2011.

[12] W. V. D. Aalst, "Service mining: using process mining to discover, check, and improve service behavior," *IEEE Transactions on Services Computing*, vol. 6, no. 4, pp. 525–535, 2013.

[13] D. R. Ferreira and D. Gillblad, "Discovering process models from unlabeled event logs," in *Proceedings of the International Conference on Business Process Management*, pp. 143–158, Springer, Ulm, Germany, September 2009.

[14] W. M. P. Van Der Aalst, B. F. Van Dongen, C. W. Günther et al., "Process mining with ProM," in *Proceedings of the 19th Belgian-Dutch Conference on Artificial Intelligence (BNAIC '07)*, pp. 453–454, Utrecht, The Netherlands, November 2007.

[15] C. W. Günther and A. Rozinat, "Disco: discover your processes," *BPM (Demos)*, vol. 940, pp. 40–44, 2012.

[16] C. W. Gunther and M. P. Wil Van Der Aalst, "Fuzzy mining-adaptive process simplification based on multi-perspective metrics," in *Business Process Management: 5th International Conference, BPM 2007, Brisbane, Australia, September 24–28, 2007. Proceedings*, vol. 4714 of *Lecture Notes in Computer Science*, pp. 328–343, Springer, Berlin, Germany, 2007.

[17] L. Cui, J. Li, and Y. Zheng, "A dynamic web service composition method based on viterbi algorithm," in *Proceedings of the IEEE 19th International Conference on Web Services (ICWS '12)*, pp. 267–271, Honolulu, Hawaii, USA, June 2012.

[18] Y. Sun, H. Yan, C. Lu, R. Bie, and Z. Zhou, "Constructing the Web of Events from raw data in the Web of Things," *Mobile Information Systems*, vol. 10, no. 1, pp. 105–125, 2014.

[19] B. Heinrich and L. Lewerenz, "Decision support for the usage of mobile information services: a context-aware service selection approach that considers the effects of context interdependencies," *Journal of Decision Systems*, vol. 24, no. 4, pp. 406–432, 2015.

[20] S. Wang, L. Sun, Q. Sun, X. Li, and F. Yang, "Efficient service selection in mobile information systems," *Mobile Information Systems*, vol. 2015, Article ID 949436, 10 pages, 2015.

[21] H. Cai, L. Cui, Y. Shi, L. Kong, and Z. Yan, "Multi-tenant service composition based on granularity computing," in *Proceedings of the 11th IEEE International Conference on Services Computing (SCC '14)*, pp. 669–676, Anchorage, Alaska, USA, July 2014.

[22] Á. Rebuge and D. R. Ferreira, "Business process analysis in healthcare environments: a methodology based on process mining," *Information Systems*, vol. 37, no. 2, pp. 99–116, 2012.

[23] W. Gaaloul, K. Baïna, and C. Godart, "Log-based mining techniques applied to web service composition reengineering," *Service Oriented Computing and Applications*, vol. 2, no. 2-3, pp. 93–110, 2008.

[24] Z. Wan, F. J. Meng, J. M. Xu, and P. Wang, "Service composition pattern generation for cloud migration: a graph similarity analysis approach," in *Proceedings of the 21st IEEE International Conference on Web Services (ICWS '14)*, pp. 321–328, Anchorage, Alaska, USA, July 2014.

[25] O. Moser, F. Rosenberg, and S. Dustdar, "Event driven monitoring for service composition infrastructures," in *Proceedings of the International Conference on Web Information Systems Engineering*, pp. 38–51, Springer, 2010.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 7538190, 17 pages
https://doi.org/10.1155/2017/7538190

Research Article

Energy-Efficient Broadcasting Scheme for Smart Industrial Wireless Sensor Networks

Zhuangbin Chen,[1] Anfeng Liu,[1] Zhetao Li,[2] Young-June Choi,[3] Hiroo Sekiya,[4] and Jie Li[5]

[1]*School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2]*College of Information Engineering, Xiangtan University, Xiangtan 411105, China*
[3]*Department of Software, Ajou University, Suwon 443749, Republic of Korea*
[4]*Graduate School of Advanced Integration Science, Chiba University, Chiba 263–8522, Japan*
[5]*Faculty of Engineering, Information and Systems, University of Tsukuba, Tsukuba 305-8573, Japan*

Correspondence should be addressed to Zhetao Li; liztchina@gmail.com

Received 19 September 2016; Revised 11 December 2016; Accepted 27 December 2016; Published 23 January 2017

Academic Editor: Qingchen Zhang

In smart Industrial Wireless Sensor Networks (IWSNs), sensor nodes usually adopt a programmable technology. These smart devices can obtain new or special functions by reprogramming: they upgrade their soft systems through receiving new version of program codes. If sensor nodes need to be upgraded, the sink node will propagate program code packets to them through "one-to-many" broadcasting, and therefore new capabilities can be obtained, forming the so-called Software Defined Network (SDN). However, due to the high volume of code packet, the constraint energy of sensor node, and the unreliable link quality of wireless network, rapidly broadcasting the code packets to all nodes in network can be a challenge issue. In this paper, a novel Energy-efficient Broadcast scheme with adjustable broadcasting radius is proposed aiming to improve the performance of network upgrade. In our scheme, the nonhotspots sensor nodes take full advantage of their residual energy caused in data collection period to improve the packet reception probability and reduce the broadcasting delay of code packet transmission by enlarging the broadcasting radius, that is, the transmitting power. The theoretical analyses and experimental results show that, compared with previous work, our approach can averagely reduce the Network Upgrade Delay (NUD) by 14.8%–45.2% and simultaneously increase the reliability without harming the lifetime of network.

1. Introduction

As one of the key components of Cyber-Physical Systems [1–4], Wireless Sensor Networks (WSNs) are emerging as a promising platform which enable a wide range of applications in both military and civilian domains [5–11]. Specifically, Industrial Wireless Sensor Networks (IWSNs) are regarded as a promising paradigm for smart industrial automation [12, 13]. In smart IWSNs, a large number of sensor nodes are deployed to detect environment events, measure the physical or chemical parameters of surroundings, and report the sensed data to the remote control center wirelessly. Based on the collected data from all these sensors, the control center can send commands to machinery actuators and trigger necessary actions [12]. Comparing with traditional industrial automation systems using wired communications, IWSN brings notable advantages including lower cost, higher flexibility, and self-organizing capability, which significantly improves the industrial efficiency and productivity [12, 14, 15].

Due to the ever increasing demand for network resources, network operators and Internet Service Providers are under constant pressure to accommodate more network bandwidth and offer better service quality via periodic network upgrade [16, 17]. With the development of smart industrial as well as Software Defined Network (SDN), today the software of sensor nodes is able to be reconfigured, which adds new features to IWSNs. The flexibility of software reconfiguration and upgrade of nodes has drawn wide attention from researchers in many application fields, like monitoring traffic information, detecting real-time conditions of petroleum

pipeline. In IWSNs, if the network needs to be upgraded, the sink node will generate code packets to be broadcasted to all sensor nodes [16, 18]. After receiving the code packet, each node compiles and executes it to gain new functions, forming a more advanced IWSN. Broadcast is a very fundamental form of communication in which nodes disseminate the same information simultaneously to all of their neighbors [19]. Given a base node with a code packet to broadcast, the aim is to propagate the packet to all nodes with a high reliability while incurring minimum latency. This problem, called minimum latency broadcast scheduling (MLBS), has been studied extensively and has been shown to be NP-hard [20]. Applications for industrial automation often have very stringent requirements on communication reliability and transmission delay [12]. Nevertheless, the harshness of industrial environments poses severe challenges on the design of energy-efficient IWSN upgrade code propagation. First, wireless channels are subject to multipath fading and interuser interference, which makes it extremely difficult to satisfy the Quality-of-Service (QoS) requirements of broadcast. Second, in realistic industrial environments, the machinery obstacles, metallic frictions, engine vibrations, and equipment noise as well as the humidity and temperature fluctuations also have adversary impacts on the reliability of end-to-end transmissions [12]. Third, in IWSNs, the program code packet needed to be broadcasted is usually of high volume, making designing an Energy-Efficient Broadcast protocol an extremely difficult problem.

Although the process of code broadcast in WSNs has been deeply studied, the broadcasting reliability and delay and energy efficiency still need to be improved. We observed a special phenomenon called "energy hole" in sensor networks [11] which leads to a very low efficiency of energy utility. To be more specific, all the nodes send data to the sink node which is situated in one side of the linear network shown as Figure 1, but the "many-to-one" data collection mode causes such imbalance: the data loads of nodes in near-sink region (hotspots) are much heavier than those in far-sink region (nonhotspots). That is because they help transmit the packets generated by the outside nodes. The operation of transmitting data is the main source of energy consumption resulting in premature death of the nodes in hotspots as well as the network [21]. Some related studies have shown that, due to the impact of energy hole, there still remains up to 90% energy in the network when it dies [11]. For broadcast operation, a large broadcasting radius can improve the link quality between nodes [16]; however, it brings about a high energy consumption because of the increased data transmitting power. Since the far-sink nodes have much energy left, we believe this part of energy can be used to enlarge their broadcasting radius. Based on the above observations, in this paper, we propose a novel code packet broadcast scheme, called Energy-efficient Broadcast (EeB) which reduces the broadcasting delay and simultaneously improves the packet transmission reliability without harming the network lifetime.

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

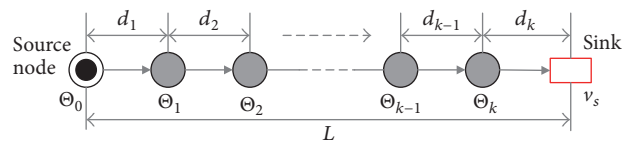


Figure 1: Linear Network Topology.

(1) We propose an Energy-efficient Broadcasting (EeB) scheme that synthetically improves the performance of packet broadcast for network upgrade which generally has three obstacles: the high volume of code packet, the constraint energy of sensor node, and the unreliable link quality between nodes. When designing the scheme, both the energy consumption in data collection stage and network upgrade stage are considered.

(2) We studied the so-called "energy hole" phenomenon [4] which causes the unbalanced utility of energy in network and then develop an algorithm that addresses the selection of code packet broadcasting radius of sensor nodes according to their residual energy. The adjustable radius allows sensor node to consume its energy more flexibly, and therefore the energy can be used to the greatest extent.

(3) The effectiveness of our scheme is evaluated in terms of Network Upgrade Delay (NUD) and packet reception probability. And the performance of the EeB comparing with a previous broadcast scheme in which all sensor nodes adopt the Same Broadcast Radius (SBR) is given in both theory and simulation.

Through our theoretical studies and a series of simulations, we demonstrate that, for the scheme proposed in this paper, packet broadcasting delay, packet transmission reliability, and energy utilization radio can be improved simultaneously. Compared with the former approach, the broadcasting delay can be averagely reduced by as much as 14.8% to 45.2%. More importantly, it improves the above performances without harming the network lifetime, which is difficult to achieve in the studies of the past.

The rest of this paper is organized as follows. In Section 2, the related works are reviewed. The system model is described in Section 3. Section 4 elaborates the design of the novel Energy-efficient Broadcast (EeB) scheme for IWSNs. The performance analyses for EeB are provided in Section 5. Section 6 is experimental results and comparisons. Finally, we conclude in Section 7.

2. Related Works

The packet broadcast of WSNs has been formulated and investigated in the literatures [20, 22, 23]. According to different applications, the existing studies can be briefly classified into the following two categories.

(1) The minimum-transmission broadcast (MTB) problem: in this kind of research, the target is to reduce the packet broadcast/transmission times of nodes. In previous studies,

nodes are assumed to be active all the time, so reducing the transmission times is to find a Minimum Connected Dominating Set (MCDS) of the network [22, 23]. The nodes in MCDS can cover the entire network, so all nodes in network can receive packet as long as the MCDS nodes broadcast packets once. In [24], the authors proved that building a minimum flooding tree is identical to finding an MCDS.

The purpose of decreasing the broadcast times of node is to reduce its energy cost, which also had been intensively studied. [25] proposed a heuristic broadcast algorithm called Broadcast Incremental Power (BIP), which constructs a broadcast tree with the broadcast source node, the root of the tree. In the process of constructing the tree, a new uncovered node will be joined into the broadcast tree at the lowest energy cost. [26] further optimized BIP algorithm and proposed a more efficient search algorithm, r-shrink, which could reduce the total broadcast energy cost by rescheduling the nonleaf nodes of the constructed broadcast tree.

In most of the WSNs, a node alternates between dormant and active states which is developed and applied to WSNs for energy conservation [7]. In broadcast mechanism, a node is required to transmit a message for multiple times to propagate the message to all of its neighbor nodes at different moments. As a result, the MTB problem in duty-cycled networks (MTB-DC problem) needs to be investigated for solutions in which both the set of forwarding nodes and their broadcast schedules are identified. Related works include Level-Based Approximation Scheme proposed by Le Duc et al. [27]. They first identified the forwarding nodes and their corresponding receivers for all time slots and then constructed a broadcast backbone by connecting these forwarding nodes to the broadcast source.

(2) The minimum latency broadcast scheduling (MLBS): in this kind of research, both decreasing the broadcasting delay and energy consumption of nodes are studied [20, 28].

Zhao et al. [20] considered MLBS in duty-cycled WSNs and presented two approximation algorithms, BS-1 and BS-2, that produce a maximum latency of at most $((\Delta - 1)TH)$ and $(13TH)$, respectively. Here, Δ is the maximum degree of nodes, T denotes the number of time slots in a scheduling period, and H is the broadcast latency lower bound obtained from the shortest path algorithm. Khiati and Djenouri [28] proposed a Broadcast over Duty-Cycle and LEACH (BOD-LEACH) protocol, which takes advantage of LEACH's energy-efficient clustering. The proposed protocol adds new common static and dynamic broadcast periods to support and accelerate broadcasting. The dynamic periods are scheduled following the past arrivals of messages and using a Markov chain model.

Another application scenario that requires demanding broadcasting delay is Vehicular ad hoc networks (VANETs) where broadcasting must be fast and reliable such that all the vehicles in a certain area can receive the message as quickly as possible to implement series applications, for instance, constructing routes to reach a given destination, cooperating for traffic management, or preventing the driver of dangers on the road [29, 30]. In [29] Gonzalez and Ramos proposed PDB, a Preset Delay Broadcast protocol with a fixed delay for vehicles attempting to retransmit a warning message, which provides a fast and reliable dissemination. They showed that, by adequately setting the waiting time for the relay candidates, the delay to cover a given area can be significantly reduced, while at the same time preserving a good reliability.

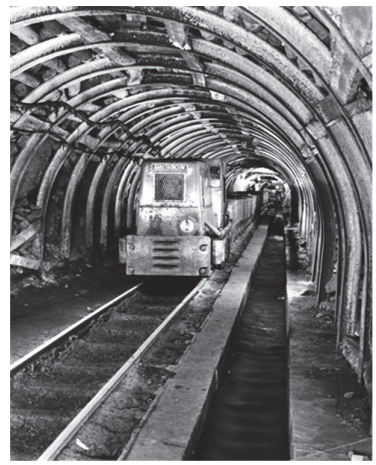


Figure 2: The haulage roadway.

3. System Model and Problem Statements

3.1. Network Model. In this paper, linear sensor network is adopted which has been studied by He et al. [9]. The linear network is a network that consists of $k + 1$ homogenous static sensor nodes and 1 sink node; that is, node set $\Theta = \{\Theta_0, \Theta_1, \Theta_2, \ldots, \Theta_k, v_s\}$ is randomly deployed on a line, as shown in Figure 1. The subscript also represents the ID number of a node; for example, the ID of node Θ_i is i. v_s represents the base station (called sink node) and is located in one side of the linear network; other nodes are common nodes, among which Θ_0 represents the first node of the network, called source node. The distance between any two adjacent nodes, say Θ_i and Θ_{i+1}, $0 \le i \le k - 1$, is denoted as d_i m. For the convenience of calculation, the distance between any two adjacent nodes will be set to a common value, r m, that is, $d_1 = d_2 = \cdots = d_k = r$. The data packet generated by any node Θ_i, $0 \le i \le k$ should be sent to the sink node by multihop route [31]. Linear sensor network is generally applied in linear application environment such as industrial production line, monitor of traffic information, surveillance of boundary line, and detection of petroleum pipeline [9].

A real-world project which applied linear topology is provided. In the mines of the Nanyang Coal Industry Co., Ltd., Hengyang, China, the main haulage roadway is approximately 12,000 m long, and most return airways have lengths of more than 1000 m [32]. This kind of underground coalmine tunnels is usually very long and narrow, and some tunnels are approximately thousands of kilometers in length but only several meters in width, as shown in Figure 2. Therefore, data transmission suffers from large delay, unbalanced energy

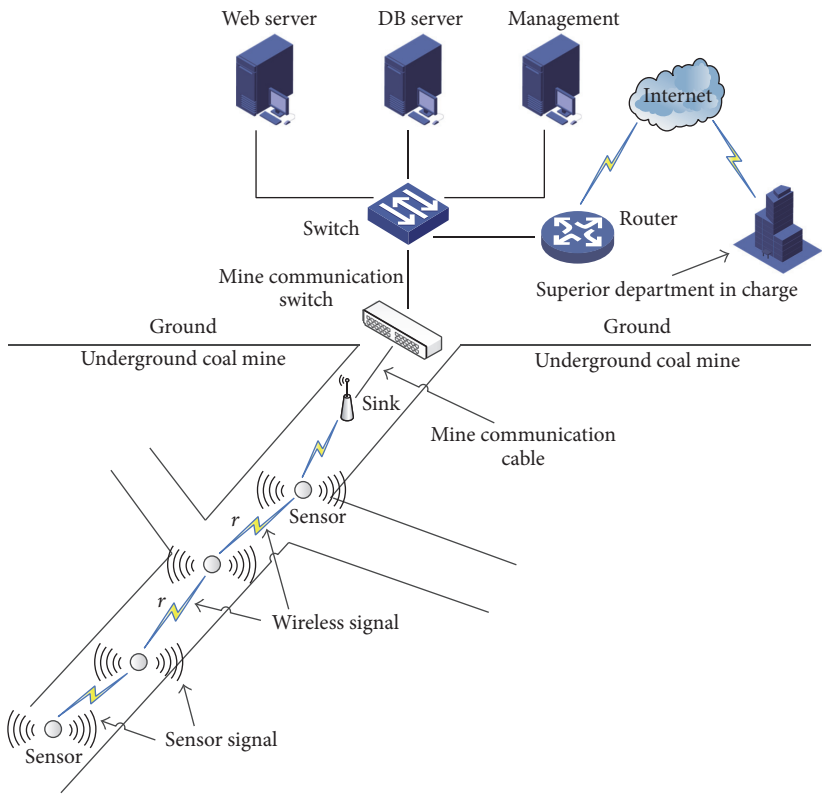


Figure 3: Architecture of line topology in the haulage roadway.

consumption, and number of retransmission because of the unreliability of wireless links.

Figure 3 shows the architecture of line topology used in the haulage roadway which contains many sensor nodes, one sink node, and one mine communication switch in the underground coalmine. In the data collection stage, the sensor node transfers sensed data (e.g., Gas concentration and CO concentration) to the sink node through multihop. All of the data are sent to the mine communication switch and are then transferred to the switch, certain types of servers, and routers on the ground. Finally, the relevant department obtains production information. On the contrary, code packets are transmitted from sink node to sensor nodes through broadcasting in the network upgrade stage.

The operation of network system is broken into rounds, and each node in network has a probability of λ to generate a packet in each data collection round. We consider a static wireless sensor network where sensor nodes do not move once deployed. The energy of common node is limited, while the energy of the sink node is infinite. Since the event production rate of node that we considered is sparse, congestions will not occur during the process of packets' routing to the sink node. Therefore, we do not take account of the queue waiting delay of packets within nodes [33].

Once the network needs to be upgraded, sink node will send program code packets to its nearby nodes through broadcasting. If the nearby nodes can successfully receive the code packet, they will also broadcast the packet to their nearby sensor nodes and forth until all nodes in network receive the code packet to carry out the upgrade process. In order to save energy, nodes will broadcast its received packet only once for avoiding too much duplicate packet retransmission. And in order to avoid the back-transmission of code packet, every node will add its ID number to the header of the code packet before broadcasting, and every time a node receives a packet, it will determine whether to broadcast it or not by comparing its own ID to the ID stored in the header of the packet. If its own ID is bigger, then this code packet will be discarded. As the radio links between nodes are imperfect, failures of transmissions always exist in the process of code packet broadcast, which will be studied in later Section 4.1. However, as EeB lays special stress on designing a better broadcast algorithm, in data collection period, the packet transmission between any two nodes will be assumed as failure-free for the convenience of calculating the residual energy of sensor nodes.

3.2. Energy Consumption Model. In this paper, we adopt the topical energy consumption model [4, 8]; that is, in wireless communications, the energy consumption of packet transmission is divided into two parts, the power consumption of the power amplifier, which can be controlled, and other circuits power consumption, while the energy used for reception is mainly the circuits power consumption. The realistic node energy model obtained from measurement results can be found in [34]. The transmission energy consumption, ω_t,

Table 1: Network parameters.

Symbol	Description	Value
d_0	Threshold distance (m)	87
r_s	Sensing range (m)	15
E_{elec}	Transmitting circuit loss (nJ/bit)	50
e_{fs}	Power amplification for the free space (pJ/bit/m^2)	10
e_{amp}	Power amplification for the multipath fading (pJ/bit/m^4)	0.0013
E_{ini}	Initial energy (J)	0.5

follows (1) and energy consumption for reception, ω_r, follows (2).

$$\begin{aligned} \omega_{t,1}(d) &= lE_{\text{elec}} + l\varepsilon_{\text{fs}}d^2, \quad \text{if } d < d_0 \\ \omega_{t,2}(d) &= lE_{\text{elec}} + l\varepsilon_{\text{amp}}d^4, \quad \text{if } d \geq d_0, \end{aligned} \tag{1}$$

$$\omega_r = lE_{\text{elec}}, \tag{2}$$

where E_{elec} represents the transmitting circuit loss; both the free-space (d^2 power loss) and the multipath fading (d^4 power loss) channel models are used; if the transmission distance is less than the threshold d_0, the power amplifier loss is based on free-space model, while if the transmission distance is larger than or equal to the threshold d_0, then the multipath attenuation model is used; ε_{fs} and ε_{amp} are the energy required by power amplification in the two models; l is the number of bits in a packet. The above parameter settings are given in Table 1, as adopted in [1, 4, 6, 8]. And for the convenience of readers to understand this paper, Notations Section summarizes the notations used in this paper.

3.3. Problem Statements. Improving the QoS of broadcast is a problem of multiple targets optimization. Some definitions are given to clearly describe the study objects of network we are trying to improve in this paper.

Definition 1. Network Upgrade Delay is denoted as D_{NUD}. D_{NUD} refers to the time duration between the generation of program code packet in the sink node and the packet to be transmitted to all nodes in network through broadcasting. Let T_{sink} stand for the time instant that code packet is generated in the sink node and T_{last} present the time instant when the last node receives the code packet; then the Network Upgrade Delay minimization can be expressed as

$$\min\left(D_{\text{NUD}}\right) = \min\left(T_{\text{last}} - T_{\text{sink}}\right). \tag{3}$$

Definition 2. Network upgrade reliability is denoted as ϕ_{NUR}. It should be guaranteed, which means it should be higher than or at least equal to the minimum reliability, $\wp$, required by applications. Let β_i stand for the broadcasting reliability of the packet at the ith hop of multihop routing from sink node to source node; then network upgrade reliability can be expressed as

$$\phi_{\text{NUR}} = \prod_{(i \in \text{route path})} \beta_i \geq \wp. \tag{4}$$

Definition 3. Network lifetime is denoted as ℓ. ℓ depends on the energy consumption speed of nodes. The energy consumption of node ς_i consists of (a) communication energy; for instance, E_t^c and E_r^c are used for transmitting and receiving data packets in data collection period, (b) upgrade energy consumption, E_t^u and E_r^u, which are used for transmitting and receiving program code packets in network upgrade period. Since the death time of the first node is defined as the lifetime [11], maximizing the lifetime is to minimize the energy consumption speed of the first node, which is expressed as the following formula, where E_{ini} represents the initial energy of node Θ_i:

$$\max(\ell) = \max \min_{1 \leq i \leq n} \left(\frac{E_{\text{ini}}}{\left(E_t^c + E_r^c + E_t^u + E_r^u\right)} \right). \tag{5}$$

Definition 4. Effective energy utilization rate is denoted as ξ_e. ξ_e refers to the ratio of energy efficiently utilized and the total energy in the network which can be expressed as a formula below. Our target is to maximize the energy utilization of the whole network. e_i in the formula stands for the energy consumption of node Θ_i:

$$\max(\xi_e) = \max\left(\frac{\left(\sum_{1 \leq i \leq n} e_i\right)}{\left(\sum_{1 \leq i \leq n} E_{\text{ini}}\right)} \right). \tag{6}$$

Obviously, the goal of EeB is to minimize the Network Upgrade Delay (NUD) and maximize the network upgrade reliability, ϕ_{NUR}, network life, ℓ, and effective energy utilization, ξ_e, which can be summarized as follows:

$$\begin{aligned} &\text{Minimize } D_{\text{NUD}}, \ \text{Maxmize } \phi_{\text{NUR}}, \ \ell, \ \xi_e, \\ &\min\left(D_{\text{NUD}}\right) = \min\left(T_{\text{last}} - T_{\text{sink}}\right) \\ &\phi_{\text{NUR}} = \prod_{(i \in \text{route path})} \beta_i \geq \wp \\ &\max(\ell) = \max \min_{1 \leq i \leq n} \left(\frac{E_{\text{ini}}}{\left(E_t^c + E_r^c + E_t^u + E_r^u\right)} \right) \\ &\max(\xi_e) = \max\left(\frac{\left(\sum_{1 \leq i \leq n} e_i\right)}{\left(\sum_{1 \leq i \leq n} E_{\text{ini}}\right)} \right). \end{aligned} \tag{7}$$

In addition, the optimization goal is transformed into an optimization problem of energy consumption and the quality of broadcasting code packets during network upgrade stage in the case of constraint lifetime. The problem is characterized as a trade-off between energy consumption and the nodes' broadcasting radius [35].

4. Main Design of EeB

In this section, Energy-efficient Broadcast (EeB) scheme will be proposed. Firstly, the packet reception probability model of EeB scheme will be introduced and then the algorithm details will be described.

(1) **Initialize:** (1) Network performs data collection for G rounds.
(1) (2) Network upgrades, i.e., the sink node broadcasts program code packets, so do other common nodes.
(3) (3) Each node stores its energy consumption for data collection, ΔE^c, and code packet broadcast, E^u_{old}, respectively.
(4) **For** node Θ_{k-1} to node Θ_0 **Do**
(5) Calculate node's residual energy, ΔE^c, using Eq. (26).
(6) Set node's new broadcast energy consumption E^u_{new} at 0.
(7) **While** node's E^u_{new} is less than $(\Delta E^c + E^u_{\text{old}})$ **Do**
(8) Enlarge node's broadcast radius by ΔR.
(9) Calculate node's E^u_{new} using Eq. (14).
(10) **End while**
(11) Reduce node's broadcast radius by ΔR.
// Make sure the broadcast energy cost is less than the residual energy.
(12) **End for**
(13) Output the broadcasting radius of every node, which can improve the network upgrade delay and reliability while the network lifetime can be guaranteed.

Algorithm 1: EeB for enlarging broadcasting radius of nodes.

4.1. Packet Reception Probability Model. The signal propagation in wireless communication is affected by various environment factors, such as obstacles, signal reflecting surface, and scatterers, which cause uncertainty in the quantification of the reception signal strength. Stojmenovic et al. [36] drove an approximate and accurate probability $\phi(d)$ for receiving a packet successfully as a function of distance d between two nodes by applying the log normal shadow fading model to represent a realistic physical layer. They provided a general relationship between the packet reception probability of receiver, $\phi(d)$, and communication distance, d, and path loss coefficient, L_r, which can be summarized as the following equation:

$$\phi(d) = \begin{cases} 1 - 0.5\left(\dfrac{d}{R}\right)^{qL_r}, & d < R \\ 0.5\left(\dfrac{2-d}{R}\right)^{qL_r}, & R \le d < 2R \\ 0, & d \ge 2R, \end{cases} \tag{8}$$

where q is a coefficient which depends on the length of the packet [36].

They pointed out that the packet reception probability remains $\phi(d) = 0.5$ constantly when the communication distance is equivalent to the transmission radius. Meanwhile, Stojmenovic et al. confirmed the correction of the above probability model through series of experiments and drew such conclusion: when the length of the packet $l = 120$ bits, $L_r \in [2, 6]$, and $q = 2$, the error of the packet reception probability calculated by the above model is within 4% compared with the actual probability.

The above model is adopted in this paper to quantify the packet reception probability of receiver, and some parameters are set as $q = 2$ and $L_r \in [2, 6]$. Suppose the transmission radius ratio $\delta = R/d$. The relation between $\phi(d)$, L_r, and δ is depicted in Figure 4. For any L_r, $\phi(d) = 0$ when $\delta \le 0.5$, and we have $\phi(d) = 0.5$ when $\delta = 1$. That is, the probability that a sensor node can successfully receive a packet is 0.5 when the transmission radius of node is equivalent to the distance between the sender and receiver node. When the distance is larger or equal to two times transmission radius, the reception probability becomes 0.

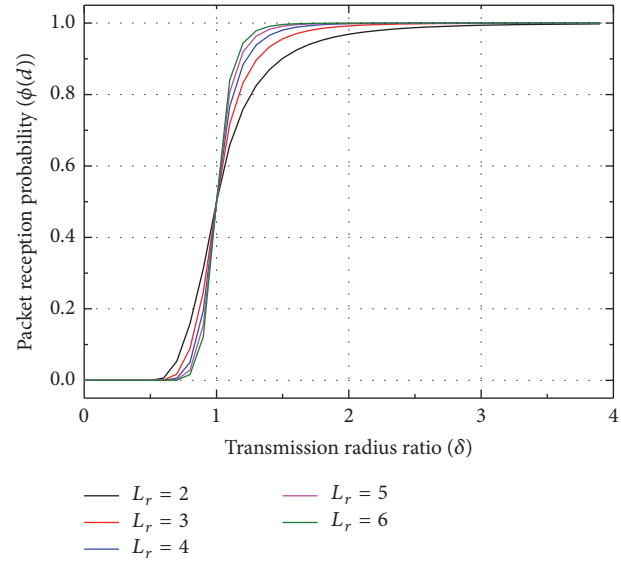


Figure 4: The expected packet reception probability versus transmission radius ratios.

4.2. EeB Algorithm. In this subsection, we propose EeB algorithm which addresses the selection of broadcasting radius of nodes according to their residual energy, as shown in Algorithm 1.

First, network performs data collection for G rounds, where G may vary in different situations to finish an event detection task. In each round, all nodes have the probability of λ to generate an event which should be delivered to the sink node by one node each hop. After that, the network upgrade will be conducted using SBR, where all nodes use the

same broadcasting radius to transmit the code packets. The broadcasting radius used in SBR is big enough to just meet the minimum reliability constraint of applications, based on which we enlarge the broadcasting radius of nodes in far-sink region to further improve the reliability. The original energy consumption for the above two steps (denoted as E^c and E^u_{old}, resp.) will be stored in each node. Finally, every node tries to enlarge its broadcasting radius by ΔR each time and calculate its new energy consumption for broadcasting; if the increased broadcast energy exceeds its residual energy, which causes a decline in network lifetime, then the radius will be reduced by ΔR; if not, the above process will be performed again. The increment ΔR actually determines how fast the radius increases. The bigger of the ΔR, the faster of the algorithm, but more residual energy will remain unused. Whereas a small ΔR can achieve a better energy utilization efficiency, the running time of the algorithm is more.

5. Theoretical Evaluation of EeB

5.1. Energy Consumption in Data Collection Stage. In this subsection, we first calculate the data load of nodes at different distances from the sink node and then provide the nodes with definite energy consumption in one round of data collection.

From Figure 1 we can see that for the node whose ID equals i, it has to receive the data packets generated by the nodes whose ID are less than i and transmit the data packets generated by nodes whose ID are less than or equal to i, that is, plus the packet generated by itself. Therefore, the total amount of packets that node Θ_i needs to receive and transmit in one data collection round is i and $i + 1$. Since the data generation rate is λ, the data loads should be multiplied by λ; that is,

$$\begin{aligned} \zeta_r^i &= i\lambda, \\ \zeta_t^i &= (i+1)\,\lambda. \end{aligned} \tag{9}$$

Corollary 5. *In EeB, for the node whose ID equals i, its total energy consumption, denoted as E_i^c, in one round of data collection can be calculated as*

$$\begin{aligned} E_i^c &= E_{i,1}^c + E_{i,2}^c \\ E_{i,1}^c &= \zeta_r^i \omega_r + \zeta_t^i \omega_{t,1}, \quad \text{if } r \le d_0 \\ E_{i,2}^c &= \zeta_r^i \omega_r + \zeta_t^i \omega_{t,2}, \quad \text{if } r > d_0 \\ \omega_{t,1} &= lE_{elec} + l\varepsilon_{fs} r^2, \\ \omega_{t,2} &= lE_{elec} + l\varepsilon_{amp} r^4 \\ \omega_r &= lE_{elec}. \end{aligned} \tag{10}$$

Proof. For the node whose ID is i, the data load it undertakes is ζ_t^i. Once a node receives a data packet, it will transmit the packet to its next node, so if the distance between two adjacent nodes is $r \le d_0$, the energy consumed for transmitting one packet is $\omega_{t,1} = lE_{\text{elec}} + l\varepsilon_{\text{fs}} r^2$, while if $r > d_0$, the energy will be $\omega_{t,2} = lE_{\text{elec}} + l\varepsilon_{\text{amp}} r^4$. Hence, the total energy used for a node to transmit data packets in data collection period is

$$\begin{aligned} E_{t,1}^i &= \zeta_t^i \omega_{t,1}, \quad \text{if } r \le d_0 \\ E_{t,2}^i &= \zeta_t^i \omega_{t,2}, \quad \text{if } r > d_0. \end{aligned} \tag{11}$$

The energy spent for a node to receive one packet is constantly $\omega_r = lE_{\text{elec}}$, and a node does not need to receive its own data packet, so the receiving data load is $\zeta_r^i = \zeta_t^i - 1$. Therefore, the total energy consumed for a node to receive data packets in a data collection round is

$$E_r^i = \zeta_r^i \omega_r. \tag{12}$$

□

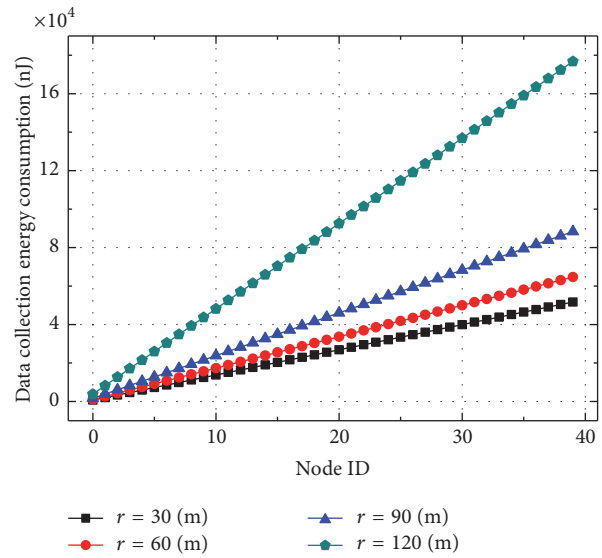


Figure 5: Data collection energy consumption of node.

Figure 5 is given to show the energy consumption of nodes for data collection versus different distances between two adjacent nodes. Obviously, the node with bigger ID consumes more energy for undertaking packets generated by nodes with smaller ID, which causes unbalance on the energy utility. And it is more uneven when r is bigger.

5.2. Energy Consumption in Nodes Upgrade Stage. In this subsection, we analyse the energy usage conditions of nodes in network upgrade stage.

Theorem 6. *In network upgrade stage, nodes will broadcast their received code packet only once; therefore, all nodes have the same receiving and transmitting packet load except for nodes in the front and tail of the linear network. We give the calculation of data load of the middle nodes, which take up*

almost the entire network, and several front and tail nodes can be computed in the same manner.

$$\xi_r^R = 2\sum_{k=1}^{\lfloor R/r\rfloor}\left(1-0.5\left(\frac{kr}{R}\right)^{qL_r}\right) + 2\sum_{k=\lfloor R/r\rfloor+1}^{\lfloor 2R/r\rfloor}\left(0.5\left(2-\frac{kr}{R}\right)^{qL_r}\right), \quad (13)$$

$$\xi_t^R = \begin{cases} 2\cdot\left\lfloor\frac{2R}{r}\right\rfloor, & \text{if node receives a packet} \\ 0, & \text{if node fails to receive a packet,} \end{cases}$$

where R is the broadcasting radius of nodes.

Proof. In Section 4.2, we know that it is possible for a node, v_i, to receive a code packet from another node, v_j, if the distance between node v_i and node v_j is less than $2R$. Hence, the total amount of nearby nodes that node v_i can receive packet from is $2\cdot\lfloor 2R/r\rfloor$. However, the reception probability from these $2\cdot\lfloor 2R/r\rfloor$ nodes should be divided into two kinds, specifically, for nodes whose distance, x, to node v_i is less than R, the probability is $1-0.5(x/R)^{qL_r}$, while if the distance is within $[R, 2R]$, then the probability becomes $0.5(2-x/R)^{qL_r}$.

For the transmitting data load, if node v_i can successfully receive a code packet, it will broadcast the packet to its $2\cdot\lfloor 2R/r\rfloor$ nearby nodes, and as node v_i will broadcast the packet only once, the data load should be $2\cdot\lfloor 2R/r\rfloor$, while if node v_i fails to receive a code packet, the transmitting data load is 0. □

Corollary 7. *For a node whose packet transmission radius is R m, the energy consumption, denoted as E_R^u, of it in network upgrade stage can be calculated as*

$$\begin{aligned} E_R^u &= E_{R,1}^u + E_{R,2}^u \\ E_{R,1}^u &= \xi_r^R\omega_r + \sum_{k=1}^{z}\omega_{t,1}^{kr}, \quad zr \le d_0 \le (z+1)r \\ E_{R,2}^u &= \xi_r^R\omega_r + \sum_{k=z+1}^{\xi_t}\omega_{t,2}^{kr}, \quad \text{if } kr > d_0 \\ \omega_{t,1}^{kr} &= lE_{elec} + l\varepsilon_{fs}(kr)^2 \\ \omega_{t,2}^{kr} &= lE_{elec} + l\varepsilon_{amp}(kr)^4 \\ \omega_r &= lE_{elec}. \end{aligned} \quad (14)$$

The proof of Corollary 7 is the same as Corollary 5.

As shown in Figure 6, the simulative results of broadcast energy consumption fit the numerical results well, and several nodes in the front and tail of network have lower energy consumption because the number of their neighbor nodes is less than $2\cdot\lfloor 2R/r\rfloor$.

5.3. Network Upgrade Reliability. To clearly show how the network upgrades, Figure 7 depicts the process of broadcasting program code packets in linear network. Each node

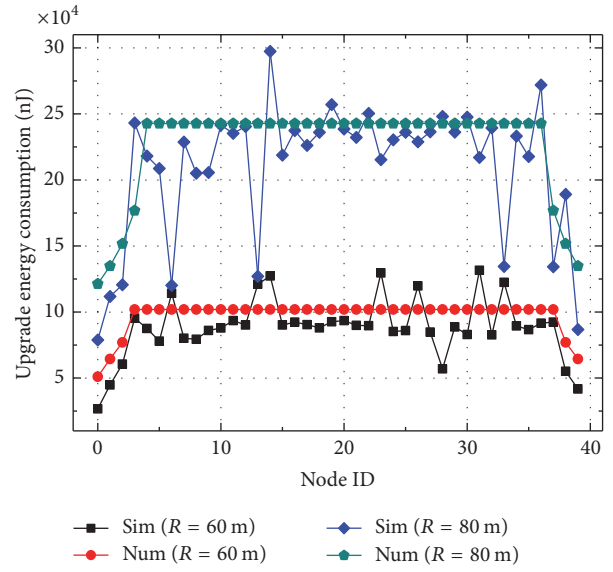


Figure 6: Upgrade energy consumption of node when $r = 40$ m.

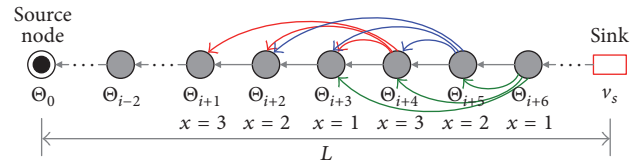


Figure 7: The process of broadcasting program code packets.

broadcasts its received code packet to the nearby nodes within its communication range ($< 2R$). In order to easily study the delay and reliability of code packet propagation, we divide all nodes in linear network into $\lfloor 2R/r-\tau\rfloor$ classes, where τ is a small enough decimal number. If the ID of a node satisfies $\mathrm{mod}(\mathrm{ID}/x) = 0$, where $1 \le x \le \lfloor 2R/r-\tau\rfloor$, then we say this node belongs to the node set of class x. The class set of nodes can be expressed as $\{x \mid \mathrm{mod}(\mathrm{ID}/\lfloor 2R/r-\tau\rfloor) + 1 = x\}$. The practical meaning of x is the number of nodes that a packet will go through by one hop in broadcast stage, which is called one-hop forwarding distance. For instance, $x = 1$ means the code packet is transmitted from sink node to source node one by one, $x = 2$ means two by two, and so forth. Note that a node can simultaneously belong to more than one class set; for example, a node whose ID is an even number belongs to class 2 set, while it also belongs to the class 1 nodes set.

Theorem 8. *The probability that a code packet can be transmitted successfully between two nodes of the same class, say x, can be calculated as*

$$\begin{aligned} p_x &= 1-\prod_{i=1}^{\lfloor 2R/r\rfloor-x}\left\{1-\phi\left[\left(\left\lfloor\frac{2R}{r}\right\rfloor-i\right)r\right]\right\} \\ 1 \le x &= \left(\mathrm{mod}\left(\frac{ID_x}{\lfloor 2R/r-\tau\rfloor}\right)+1\right) \le \left\lfloor\frac{2R}{r}-\tau\right\rfloor, \end{aligned} \quad (15)$$

where τ is a decimal which is small enough. Subtracting τ is because when $2R$ is exactly an integral multiple of r, the packet reception probability of number $2R/r$ node is 0; see Section 4.1, which results in that infinite hops are needed to transmit a code packet from the sink node to the source node, so this case should be excluded.

Proof. All nodes in network are divided into $\lfloor 2R/r - \tau \rfloor$ classes. During the code packets broadcast stage, nodes of different classes have different packet reception probabilities in each broadcast hop. For a class x node, say Θ_x, it can receive the code packets from another node Θ_i whose distance to it is less than or equal to $(\lfloor 2R/r \rfloor - x)r$; thus, its final packet reception probability should be

$$p_x = 1 - \prod_{i=1}^{\lfloor 2R/r \rfloor - x} \left\{ 1 - \phi \left[\left(\left\lfloor \frac{2R}{r} \right\rfloor - i \right) r \right] \right\}, \quad (16)$$

where the packets received from the nodes of smaller ID are excluded.

An example for better explaining the above idea is given in Figure 7. When $\lfloor 2R/r \rfloor = 4$, nodes are divided into 3 classes. Suppose Θ_{i+4}, Θ_{i+5}, and Θ_{i+6} are classes 3, 2, and 1 node, respectively, and they simultaneously broadcast their received packets. Packets from Θ_{i+4} can reach Θ_{i+1}, which is also a class 3 node, and similarly, the farthest node that the other two nodes can reach is the one with the same class. In this broadcast hop, node Θ_{i+3} $(x = 1)$ has three chances to receive packets from three nodes with larger ID, that is, Θ_{i+4}, Θ_{i+5}, and Θ_{i+6}. While Θ_{i+2} $(x = 2)$ can only receive packets from two nodes, that is, Θ_{i+4} and Θ_{i+5}, and so on. In the next broadcasting hop, node Θ_{i+2} will have a new chance to receive code packet from node Θ_{i+3}, making it become a class 1 node. Similarly, node Θ_{i+1} $(x = 3)$ becomes a class 2 node, and a new node Θ_{i-2} will become a class 3 node for having one chance to receive packet from node Θ_{i+1}. □

The packet reception probability of packet delivered by nodes of the same class is shown in Figure 8 indicating that nodes have higher reception probabilities with the increment of broadcasting radius. And the small-class nodes have relatively high reliabilities for having short transmission distances.

5.4. Network Upgrade Delay. We define the average transmission hops needed to transmit the program code packets from sink node to source node through nodes of the same classes as Network Upgrade Delay (NUD). Obviously, the NUD describes how fast the network can be thoroughly upgraded by a certain class of nodes.

Theorem 9. *For the class x nodes, their Network Upgrade Delay (NUD) can be calculated as*

$$NUD(x) = \frac{\lceil L/xr \rceil}{p_x}, \quad (17)$$

where p_x can be calculated by (15).

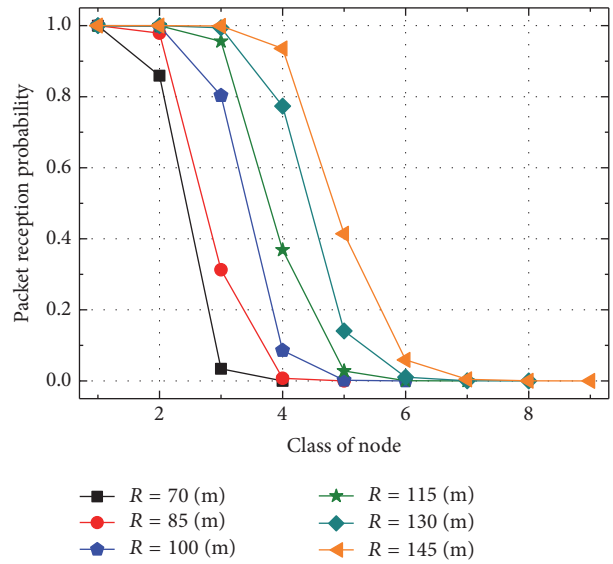


Figure 8: Packet reception probability of different-class nodes when $r = 30$ m.

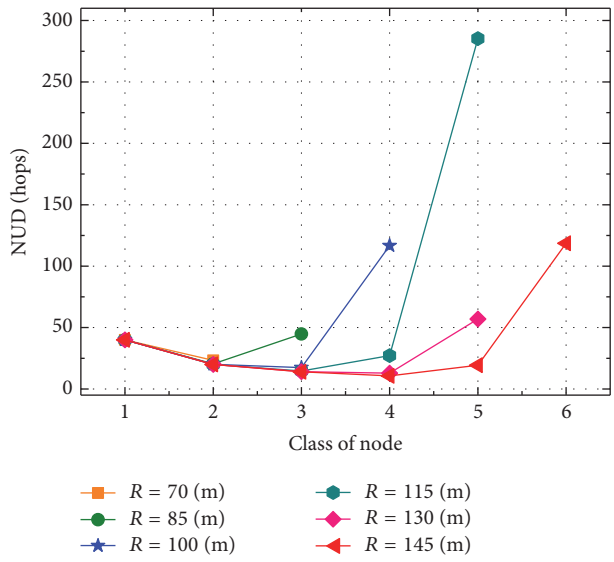


Figure 9: NUD of different-class nodes when $r = 30$ m.

Proof. For nodes of class x, $\lceil L/xr \rceil$ broadcast hops are needed to transmit code packets from sink node to source node if they do not take the reception failure into consideration. And one-hop reception reliability of p_x yields the average broadcast hops as

$$\mathrm{NUD}(x) = \frac{\lceil L/xr \rceil}{p_x}. \quad (18)$$

□

Figure 9 is given to show the NUD of nodes with different r and R settings. After eliminating some classes of extreme large NUD, we can see that middle-class nodes

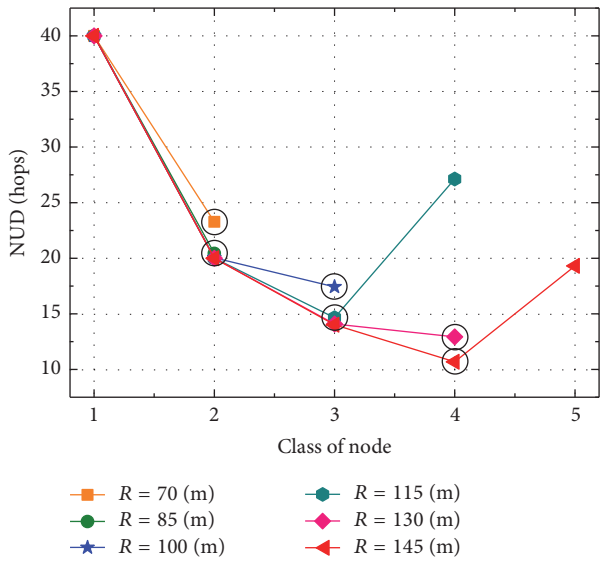


FIGURE 10: NUD of different-class nodes when $r = 30$ m.

have relative small NUD compared with nodes of smaller and bigger classes. And it is easy to explain the above phenomenon: although small-class nodes have high packet reception reliability, their one-hop forwarding distance is too short, so more hops are needed; while big-class nodes have long one-hop forwarding distance, failure of packet reception always exists, so more hops are also needed. Therefore, there must be a trade-off between them.

In addition, to clearly show the exact value of NUD of different-class nodes, we give Figure 10 by zooming in Figure 9, where the shortest NUD of different-class nodes are circled. The larger the broadcasting radius is, the less the broadcast hops are needed for all nodes in network to receive code packets, which means the Network Upgrade Delay is shorter.

In our scheme, we try to enlarge the broadcasting radius of the far-sink nodes, and generally, the further a node is from the sink node, the larger the radius it has. Since different nodes have different broadcasting radii, all nodes in network cannot be simply divided into $\lfloor 2R/r-\tau \rfloor$ classes. To obtain the actual value of NUD of EeB, we need to calculate the one-hop broadcasting delay.

Theorem 10. *For a node Θ_i, suppose its broadcasting radius is Rm, then its average one-hop broadcasting delay can be obtained as follows:*

$$d_b^R = \frac{1}{p_x}$$

$$\text{subject to } \min \frac{1}{x \cdot p_x} \tag{19}$$

$$1 \le x \le \left\lfloor \frac{2R}{r} - \tau \right\rfloor,$$

where p_x can be calculated by (15), and x is the one-hop forwarding distance, which equals the difference of the ID between node Θ_i and Θ_j whose ID meets $i - j = x$.

Proof. The code packets broadcasted from node Θ_i can be received by $\lfloor 2R/r - \tau \rfloor$ nodes on its left side, and these nodes have different packet reception probabilities which can be calculated by (15). The average one-hop broadcasting delay is the hops needed for node Θ_j to successfully receive a code packet from node Θ_i, where node Θ_j is supposed to be the node which can deliver the code packet to the source node as quick as possible making a minimal NUD. Therefore the one-hop broadcasting delay should balance the one-hop forwarding distance and packet reception probability. Since the reception probability is p_x, the hop count is $1/p_x$. □

Figure 11 is given to show the average one-hop broadcasting delay, from which we can see nodes with different broadcasting radii having very similar one-hop delay. Note that a low one-hop broadcasting delay does not mean a low NUD because the number of broadcasting hops is another important factor of NUD. Since in EeB nodes of different distances to the sink node have different broadcasting radii, combining them, we can get the actual value of NUD of EeB, so the following theorem is deduced.

Theorem 11. *Suppose a linear network is composed of $k + 1$ sensor noses and the distance between any two adjacent nodes is rm, then in EeB, the Network Upgrade Delay (NUD) can be calculated as*

$$NUD_{QIB}(k+1) = \sum_{i=x_s}^{k} \left(d_b^{R_{k+1-i}},\ i = i + x_{k+1-i} \right), \tag{20}$$

where $d_b^{R_k}$ stands for the expectation one-hop delay of the packet at the node with ID = k, whose broadcasting radius is R_k, x_k denotes the selected x by the node with ID = k using (19), and s denotes the sink node v_s especially.

Proof. Since nodes have different packet broadcasting radii, the actual value of NUD of EeB is the sum of each different one-hop delay, and each one-hop forwarding distance is also different but can be determined by (19). In broadcast operation, packets pass through several nodes each hop instead of one by one, so we use $i = i + x_{k+1-i}$ to show the route path of packets. □

Figure 12 shows the Network Upgrade Delay. The index of x-axis (original broadcasting radius) means the common radius adopted by all nodes in SBQ, which is also the start point for EeB to increase.

Theorem 12. *In linear network, the weighted one-hop broadcasting delay (denoted as D_w^{k+1}), which reflects the universality of the reduced delay with EeB approach, can be calculated as*

$$D_w^{k+1} = \frac{1}{k+1} \cdot \sum_{n=0}^{k} d_b^{R_k}, \tag{21}$$

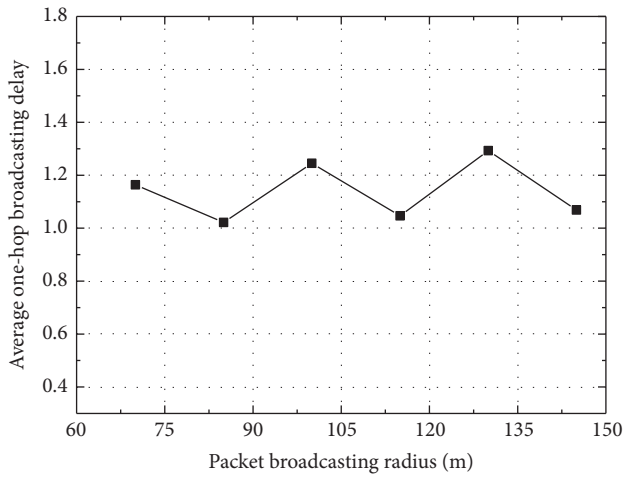


Figure 11: Average one-hop broadcasting delay.

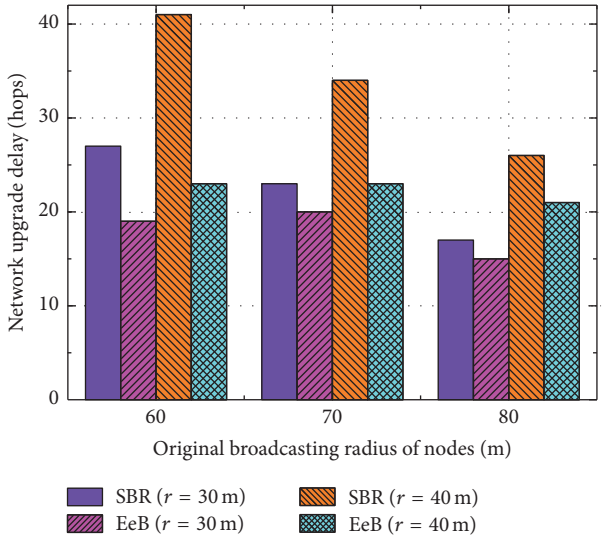


Figure 12: Network Upgrade Delay.

where the variables are the same as the corresponding items in Theorem 11.

Proof. The total number of nodes in linear network is $k + 1$; therefore for a node Θ_i, the one-hop broadcasting delay of the network upgrade can be expressed as $(1/(k+1))d_b^{R_i}$. Integrally, to the entire linear network, the weighted delay of one-hop code packet broadcast is as follows:

$$D_w^k = \frac{1}{k+1} \cdot \sum_{n=0}^{k} d_b^{R_k}. \tag{22}$$

□

Figure 13 is given to show the weighted one-hop broadcasting delay of network.

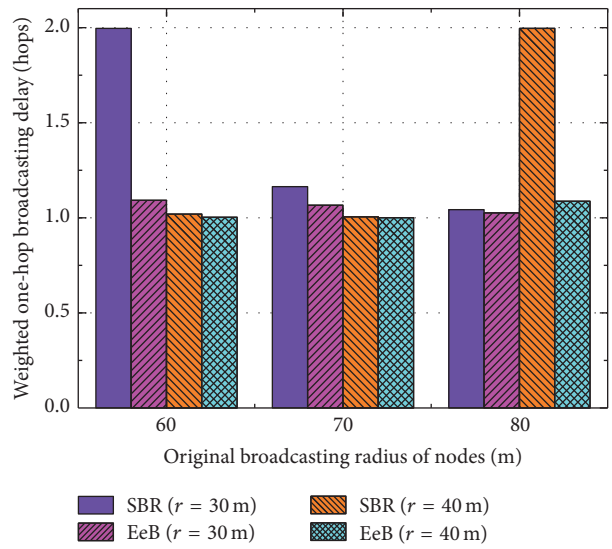


Figure 13: Weighted one-hop broadcasting delay.

5.5. Energy Utilization Rate of Network. The innovation of EeB protocol is using the residual energy of far-sink nodes to enlarge their code packet broadcasting radius, by which the QoS of network update is improved as well as the energy utilization rate. In this subsection, we evaluate the performance of EeB in terms of the enhanced energy usage rate.

Theorem 13. *In EeB protocol, the energy utilization rate of network containing $k + 1$ sensor nodes can be calculated as*

$$\xi_e = \frac{\sum_{i=0}^{k} \left(E_i^c + E_i^u\right)}{\sum_{i=0}^{k} E_{ini}^i} \times 100\%, \tag{23}$$

where E_i^c is the energy used for collecting sensed data and E_i^u is spent in broadcasting code packets by node Θ_i whose initial energy is E_{ini}^i.

Proof. According to Definition 4 in Section 3.3, the energy utilization rate of network is the ratio of energy efficiently utilized and the total energy in the network. Each node has two parts of energy consumption, that is, collecting data and broadcasting code packets, respectively, so the above equation is obtained. □

Figure 14 displays the situations of energy utilization rate when the network dies. It can be seen that, in EeB, at least 50% energy has been effectively utilized by sensor nodes, illustrating that a great energy efficiency has been made by EeB compared with SBR.

5.6. Network Lifetime. In this subsection, we provide the calculation of network lifetime which can be maintained in EeB protocol because the increased energy for broadcasting code packets is always kept below or equal to the residual energy of nodes.

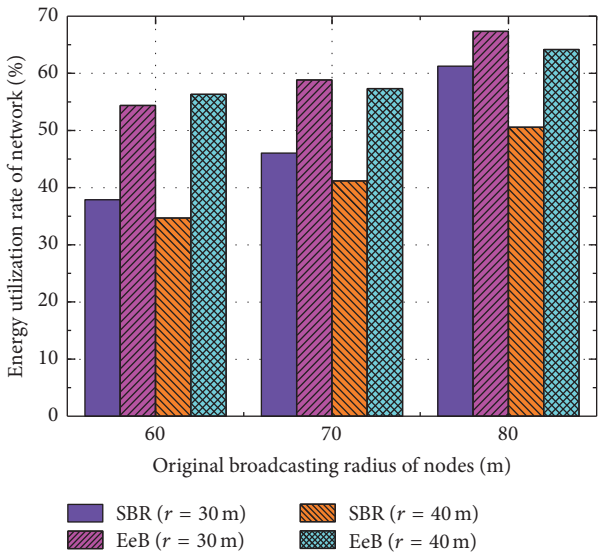


Figure 14: Energy utilization rate of network.

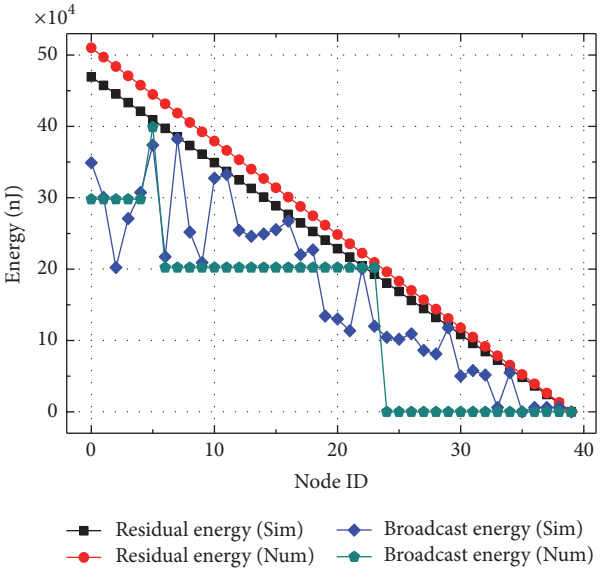


Figure 15: The increased energy used for enlarging broadcasting radius of node when r = 30 m and R = 60 m.

Theorem 14. *Consider that the initial energy of each node in network is E_{ini}; in EeB, the lifetime (denoted as ℓ) of network can be obtained as*

$$\ell = \frac{E_{ini}}{E^c_{max} + E^u_{max}}, \tag{24}$$

where E^c_{max}, E^u_{max} are the energy consumption of the node consuming the most energy in data collection stage and network update stage, respectively.

Proof. The node consuming the most energy dies first, and its energy consumption can be calculated as $(E^c_{\max} + E^u_{\max})$. Since the death time of the first node is defined as the lifetime of network [11], the lifetime of network can be obtained by

$$\ell = \frac{E_{\text{ini}}}{E^c_{\max} + E^u_{\max}}. \tag{25}$$

□

6. Experimental Evaluation of EeB

In this section, we evaluate the experimental performance of Energy-efficient Broadcast (EeB) scheme in terms of energy consumption and NUD compared with SBR in which all nodes in network adopt the same broadcasting radius. OMNET++ is employed for simulative verification [37]. Without loss of generality, the network parameters are as follows: event generation probability of node in data collection period is λ = 0.1 and 40 nodes are randomly deployed in line with 1 sink node at one side; after G = 10 rounds of data collections the network upgrade will be conducted, and the increment of broadcasting radius is ΔR = 1 m. Other parameter settings will be particularly indicated.

6.1. Energy Consumption of EeB. Since the close-sink nodes have to undertake the packets generated by far-sink nodes, the close-sink nodes consume more energy, which causes unbalance of energy usage in network. The remaining energy of far-sink nodes is called residual energy, and the residual energy of node with ID = i, that is, Θ_i, can be calculated as

$$\Delta E^c_i = E^c_{\max} - E^c_i, \tag{26}$$

where $E^c_{\max}$ is the energy consumption of the node which consumes the most energy and E^c_i is the energy consumption of node Θ_i; both are during the data collection period.

After enlarging the broadcasting radius, a node will have a new energy cost for broadcasting packets, and this increased broadcasting energy can be computed as

$$\Delta E^u_i = E^u_{\text{new},i} - E^u_{\text{old},i}, \tag{27}$$

where $E^u_{\text{new},i}$ and $E^u_{\text{old},i}$ are the new and old broadcasting energy cost of node Θ_i, respectively.

Figures 15 and 16 are given to show the simulative and theoretical energy consumption of nodes. We can see that the further the node is from the sink node, the more the residual energy it has, which can be used to enlarge its broadcasting radius. In EeB, node's ΔE^u always remains lower than its ΔE^c, so the lifetime of network can be maintained. The corresponding enlarged broadcasting radius of nodes is shown in Figures 17 and 18, respectively.

6.2. Energy Utilization Rate of Network. Figure 19 shows the simulative results of energy utilizations of network which are consistent with the previous theoretical analyses shown in Figure 14. The corresponding reduced ratio of energy usage rate is presented in Figure 20 illustrating that at least 20% of improvement has been made.

6.3. Network Upgrade Delay. Figure 21 presents the simulative results of the average one-hop broadcasting delay.

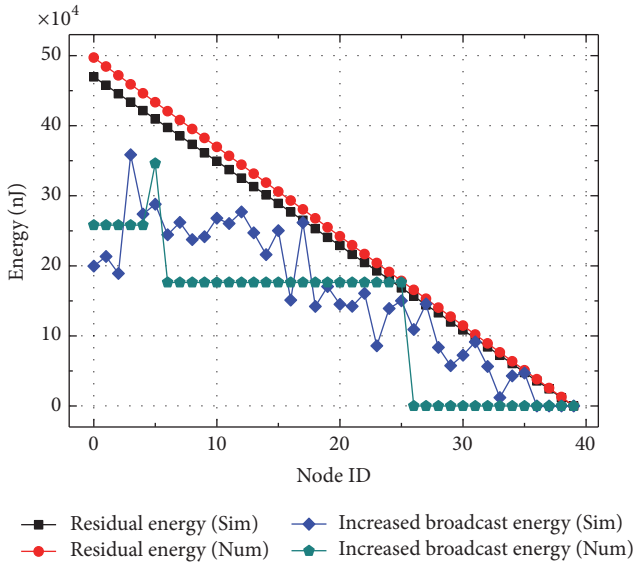


FIGURE 16: The increased energy used for enlarging broadcasting radius of node when $r = 40$ m and $R = 60$ m.

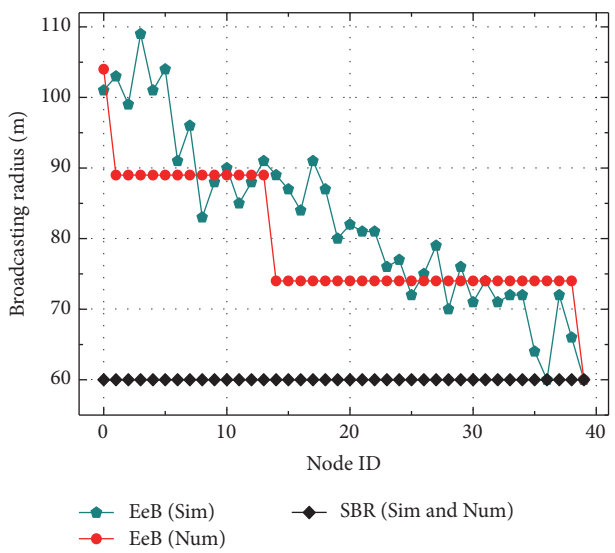


FIGURE 17: Broadcasting radius of node when $r = 30$ m and $R = 60$ m.

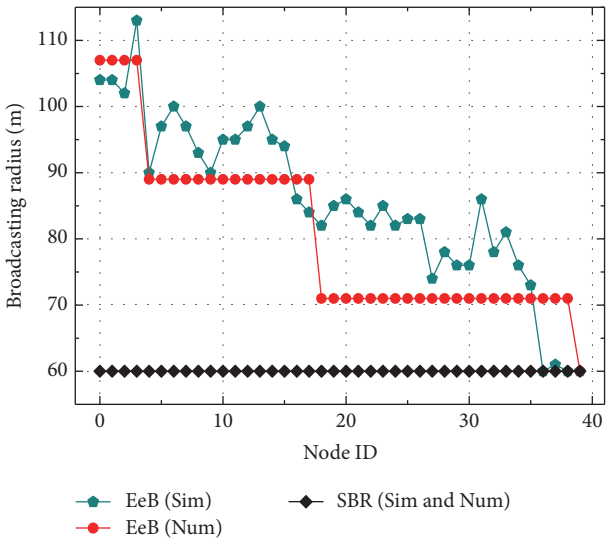


FIGURE 18: Broadcasting radius of node when $r = 40$ m and $R = 60$ m.

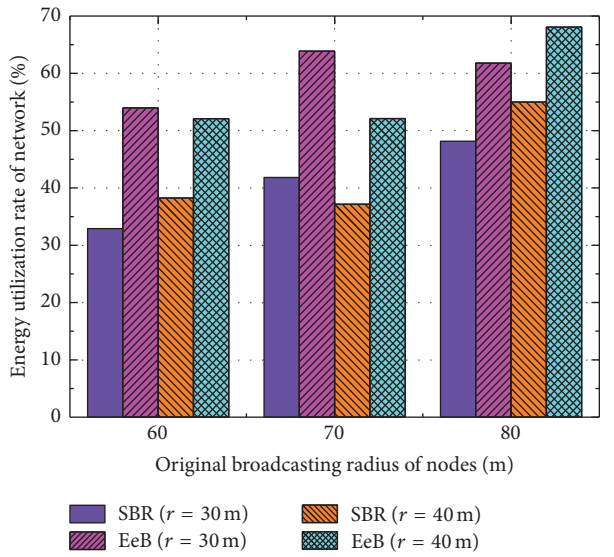


FIGURE 19: Energy utilization rate of network.

Generally, the delay of EeB is lower than SBR which indicates the great efficiency of EeB in improving the network upgrade speed. The delays of middle nodes are relatively larger and vary a lot.

We also simulated the actual hops needed for the entire network to be upgraded, that is, the time duration from the generation of program code packet in sink node to all sensor nodes in network receiving code packets as shown in Figure 22. The one hop of broadcast is defined as all sensor nodes owning code packets broadcast the packet to their nearby nodes. The delay of network upgrade is measured by the number of broadcasting hops. Specifically, the Network Upgrade Delay is enhanced by 14.8%–45.2% as shown in Figure 23. The bigger the transmission radius ratio (r/R), the larger the reducing rate.

6.4. Network Upgrade Reliability. The packet reception reliability is evaluated in this subsection. In the simulation, we recorded the average amount of code packets received by nodes in the network upgrade stage, and two experimental results are given in Figure 24. In the first experiment, the average distance between two adjacent nodes was set to be 30 m and the broadcasting radius used in SBR was 60 m. Significantly more packets were successfully received, which demonstrates a higher reliability of EeB. Similar situation can be seen in the second experiment, where the average distance between two nodes was 40 m and the broadcasting radius

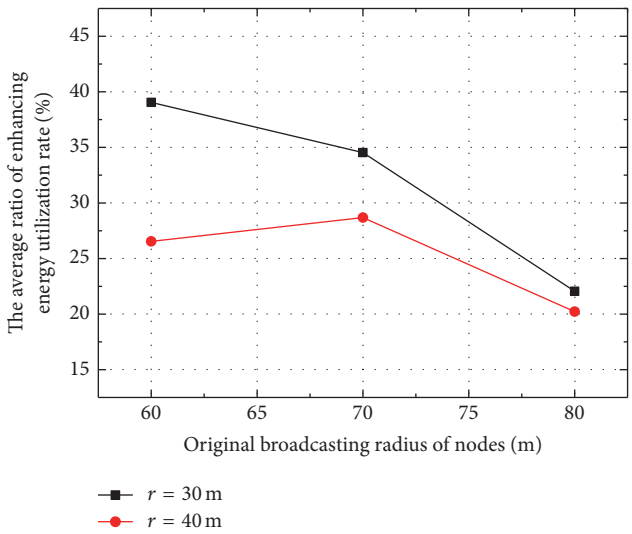


Figure 20: The average ratio of enhancing energy utilization rate.

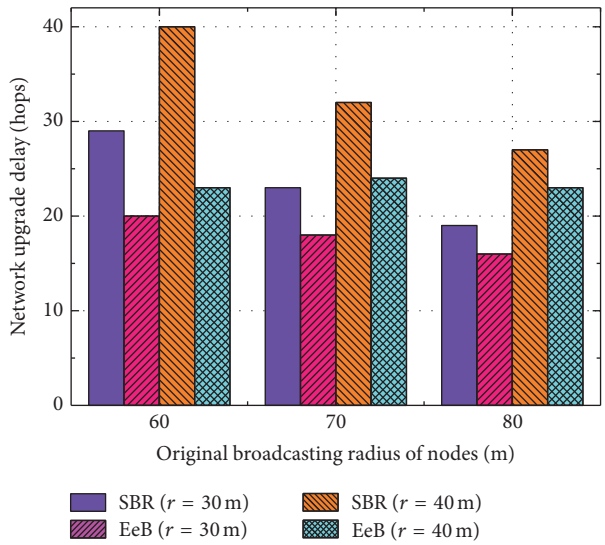


Figure 22: Simulative Network Upgrade Delay.

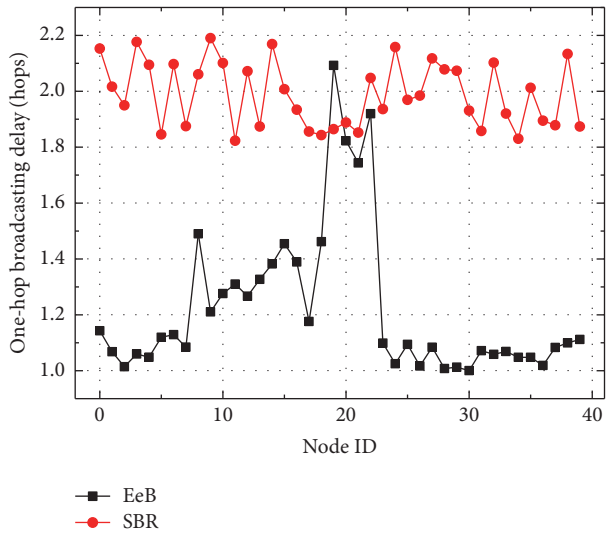


Figure 21: Average one-hop broadcasting delay of node when r = 30 m and R = 60 m.

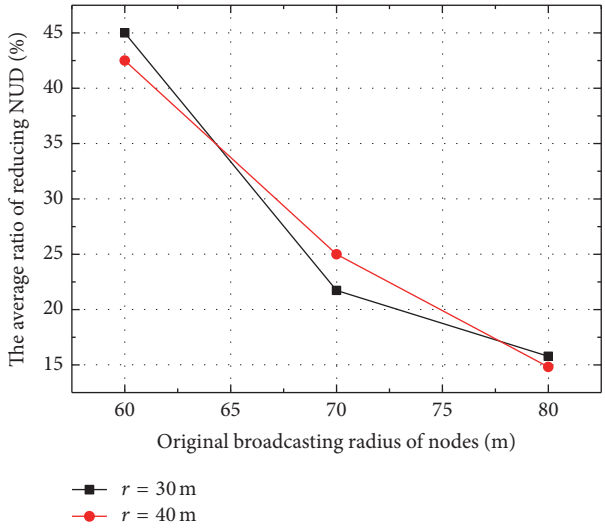


Figure 23: The average ratio of reducing NUD.

remained the same. In case 2, many nodes failed to receive a packet, while EeB had a better performance.

6.5. The Effect of Parameters. Parameter λ represents the probability of generating an event in every data collection round. It is determined by the number of nodes and detection targets of the network. In general, the bigger λ is, the more the energy the nodes consume for delivering more data packets in data collection stage, as shown in Figure 25. In this way, the parameter λ will affect the network lifetime. Figure 26 shows the different lifetime under different λ (0.08, 0.10, 0.12, and 0.15). The simulative results show that the greater λ, the shorter the network lifetime, which are consistent with the theoretical analyses. In addition, the lifetimes of EeB are almost the same as that of SBR, which indicates the original lifetime can be guaranteed in EeB.

7. Conclusion

In smart Industrial Wireless Sensor Networks (IWSNs) in which sensor nodes adopt a programmable technology, it is an important issue to upgrade the software of sensor nodes with high reliability and low latency. Generally, the network upgrade process starts when sink node propagates program code packets to its nearby nodes, and most of the existing studies focused on constructing spinning tree and reasonable dispatching time slot to reduce packet broadcasting delay and energy consumption of nodes. Different from them, the Energy-efficient Broadcast (EeB) scheme proposed in this paper adopts a novel strategy to enlarge the packet transmission radius of nodes in far-sink region using their residual energy caused in data collection period. Hence, the packet broadcasting reliability and delay can be simultaneously improved.

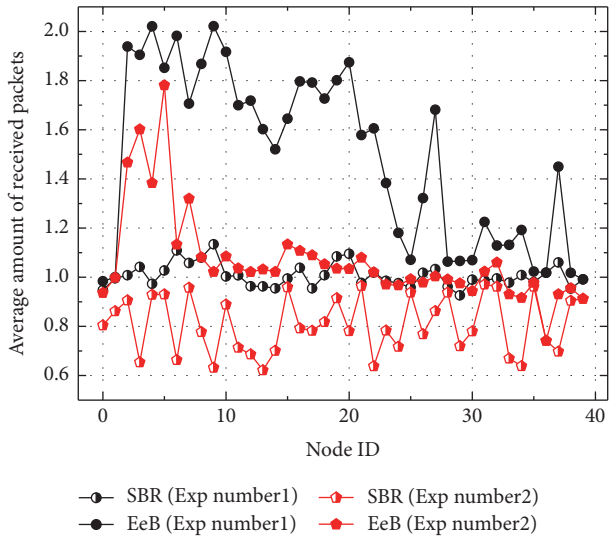


Figure 24: Average amount of packets received by node.

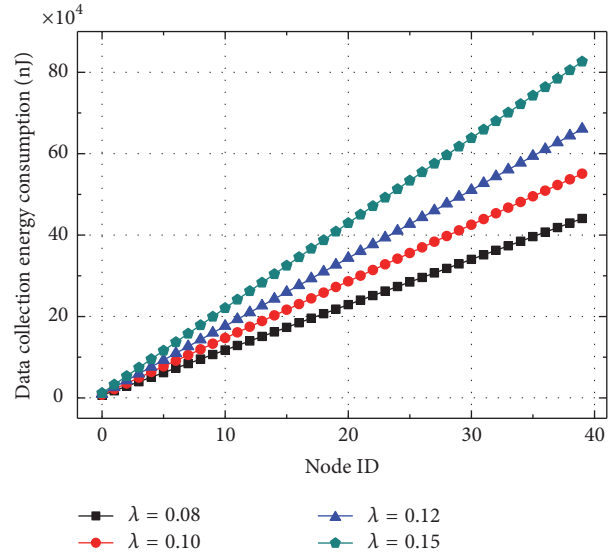


Figure 25: Data collection energy consumption versus different λ.

The EeB scheme addresses efficiently using the residual energy; however, the broadcasting behaviour of sensor nodes is too naive. We simply assume that nodes broadcast their packets only once for avoiding too much duplicate packet retransmission. However, some redundant packets retransmissions still exist because nodes are unable to know whether or not their nearby nodes have received the codes packets, so they will do the unnecessary packet transmission. In further work, we are trying to fill this gap by redesigning the behaviour of nodes to make them appreciable.

Notations

ω_r: The power used for receiving a packet
ω_t: The power used for transmitting a packet
ζ_t: The transmitting packets load of a node during data collection period
ζ_r: The receiving packets load of a node during data collection period
ς_t: The transmitting packets load of a node during network upgrade period
ς_r: The receiving packets load of a node during network upgrade period
r: The common distance between any two adjacent nodes in linear network (m)
R: The broadcasting radius of nodes (m)
λ: Event production rate.

Competing Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61379110, 61073104, 61379115, 61311140261, 61572528, 61272494, and 61572527) and the National Basic Research Program of China (973 Program) (2014CB046305).

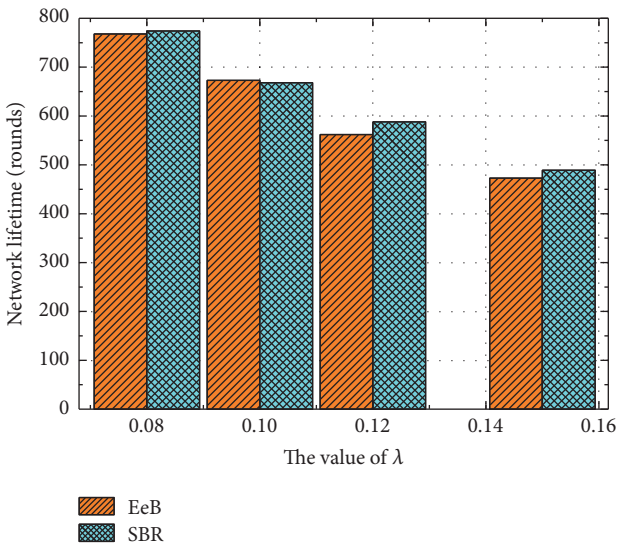


Figure 26: Network lifetime versus different λ.

References

[1] X. Liu, M. Dong, K. Ota, P. Hung, and A. Liu, "Service pricing decision in cyber-physical systems: insights from game theory," *IEEE Transactions on Services Computing*, vol. 9, no. 2, pp. 186–198, 2016.

[2] S. He, D. Shin, J. Zhang, J. Chen, and Y. Sun, "Full-view area coverage in camera sensor networks: dimension reduction and near-optimal solutions," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7448–7461, 2016.

[3] S. He, J. Chen, X. Li, X. S. Shen, and Y. Sun, "Mobility and intruder prior information improving the barrier coverage of sparse sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1268–1282, 2014.

[4] A. Liu, X. Jin, G. Cui, and Z. Chen, "Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network," *Information Sciences*, vol. 230, pp. 197–226, 2013.

[5] 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.

[6] X. Liu, "A deployment strategy for multiple types of requirements in wireless sensor networks," *IEEE Transactions on Cybernetics*, vol. 45, no. 10, pp. 2364–2376, 2015.

[7] A. Liu, M. Dong, K. Ota, and J. Long, "PHACK: an efficient scheme for selective forwarding attack detection in WSNs," *Sensors*, vol. 15, no. 12, pp. 30942–30963, 2015.

[8] M. Dong, K. Ota, A. Liu, and M. Guo, "Joint optimization of lifetime and transport delay under reliability constraint wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 1, pp. 225–236, 2016.

[9] S. He, J. Chen, F. Jiang, D. K. Y. Yau, G. Xing, and Y. Sun, "Energy provisioning in wireless rechargeable sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 1931–1942, 2013.

[10] A. Liu, X. Liu, H. Li, and J. Long, "MDMA: a multi-data and multi-ACK verified selective forwarding attack detection scheme in WSNs," *IEICE Transactions on Information and Systems*, vol. E99D, no. 8, pp. 2010–2018, 2016.

[11] 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.

[12] L. Sun, P. Ren, Q. Du, and Y. Wang, "Fountain-coding aided strategy for secure cooperative transmission in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 291–300, 2016.

[13] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, and X. Shen, "EPPDR: an efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053–2064, 2014.

[14] Z. Tang, A. Liu, Z. Li, Y. Choi, H. Sekiya, and J. Li, "A trust-based model for security cooperating in vehicular cloud computing," *Mobile Information Systems*, vol. 2016, Article ID 9083608, 22 pages, 2016.

[15] H. Li, D. Liu, Y. Dai, and T. H. Luan, "Engineering searchable encryption of mobile cloud networks: when QoE meets QoP," *IEEE Wireless Communications*, vol. 22, no. 4, pp. 74–80, 2015.

[16] Y. Xu, A. Liu, and C. Changqin, "Delay-aware program codes dissemination scheme in internet of everything, mobile information systems," *Mobile Information Systems*, vol. 2016, Article ID 2436074, 18 pages, 2016.

[17] J. Xiao and R. Boutaba, "Customer-centric network upgrade strategy: maximizing investment benefits for enhanced service quality," in *Proceedings of the IEEE/IFIP Network Operations and Management Symposium: Managing Next Generation Convergence Networks and Services (NOMS '04)*, pp. 759–772, Seoul, Korea, April 2004.

[18] X. Zheng, J. Wang, W. Dong, Y. He, and Y. Liu, "Bulk data dissemination in wireless sensor networks: analysis, implications and improvement," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1428–1439, 2016.

[19] B. Rashid, M. H. Rehmani, and A. Ahmad, "Broadcasting strategies for cognitive radio networks: taxonomy, issues, and open challenges," *Computers & Electrical Engineering*, vol. 52, pp. 349–361, 2016.

[20] D. Zhao, K.-W. Chin, and R. Raad, "Approximation algorithms for broadcasting in duty cycled wireless sensor networks," *Wireless Networks*, vol. 20, no. 8, pp. 2219–2236, 2014.

[21] Y. Liu, M. Dong, K. Ota, and A. Liu, "ActiveTrust: secure and trustable routing in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 2013–2027, 2016.

[22] B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit disk graphs," *Discrete Mathematics*, vol. 86, no. 1-3, pp. 165–177, 1990.

[23] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

[24] H. Lim and C. Kim, "Flooding in wireless ad hoc networks," *Computer Communications*, vol. 24, no. 3-4, pp. 353–363, 2001.

[25] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the construction of energy-efficient broadcast and multicast trees in wireless networks," in *Proceedings of the IEEE 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, IEEE Press, Tel Aviv, Israel, February 2000.

[26] A. K. Das, R. J. Marks, M. El-Sharkawi, P. Arabshahi, and A. Gray, "r-shrink: a heuristic for improving minimum power broadcast trees in wireless networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '03)*, San Francisco, Calif, USA, December 2003.

[27] T. Le Duc, D. T. Le, V. V. Zalyubovskiy, D. S. Kim, and H. Choo, "Level-based approach for minimum-transmission broadcast in duty-cycled wireless sensor networks," *Pervasive and Mobile Computing*, vol. 27, pp. 116–132, 2016.

[28] M. Khiati and D. Djenouri, "BOD-LEACH: broadcasting over duty-cycled radio using LEACH clustering for delay/power efficient dissimilation in wireless sensor networks," *International Journal of Communication Systems*, vol. 28, no. 2, pp. 296–308, 2015.

[29] S. Gonzalez and V. Ramos, "Preset delay broadcast: a protocol for fast information dissemination in vehicular ad hoc networks (VANETs)," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, article 117, pp. 1–13, 2016.

[30] X. Liu and G. Yan, "Analytically modeling data dissemination in vehicular ad hoc networks," *Ad Hoc Networks*, vol. 52, pp. 17–27, 2016.

[31] J. Gui and Z. Zeng, "Joint network lifetime and delay optimization for topology control in heterogeneous wireless multi-hop networks," *Computer Communications*, vol. 59, pp. 24–36, 2015.

[32] X. Xia, Z. Chen, H. Liu, H. Wang, and F. Zeng, "A routing protocol for multisink wireless sensor networks in underground coalmine tunnels," *Sensors*, vol. 16, no. 12, p. E2032, 2016.

[33] K. P. Naveen and A. Kumar, "Relay selection for geographical forwarding in sleep-wake cycling wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 3, pp. 475–488, 2013.

[34] K. Xu and I. Howitt, "Realistic energy model based energy balanced optimization for low rate WPAN network," in *Proceedings of the IEEE Southeastcon*, pp. 261–266, Atlanta, Ga, USA, March 2009.

[35] A. Liu, X. Liu, and J. Long, "A trust-based adaptive probability marking and storage traceback scheme for WSNs," *Sensors*, vol. 16, no. 4, article no. 451, 2016.

[36] I. Stojmenovic, A. Nayak, J. Kuruvila, F. Ovalle-Martinez, and E. Villanueva-Pena, "Physical layer impact on the design and performance of routing and broadcasting protocols in ad hoc and sensor networks," *Computer Communications*, vol. 28, no. 10, pp. 1138–1151, 2005.

[37] OMNet++ Network Simulation Framework, http://www.omnetpp.org/.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 1248578, 14 pages
https://doi.org/10.1155/2017/1248578

Research Article

Exploiting Wireless Received Signal Strength Indicators to Detect Evil-Twin Attacks in Smart Homes

Zhanyong Tang,[1] Yujie Zhao,[1] Lei Yang,[1] Shengde Qi,[1] Dingyi Fang,[1] Xiaojiang Chen,[1] Xiaoqing Gong,[1] and Zheng Wang[2]

[1]*School of Information Science and Technology, Northwest University, Xi'an, China*
[2]*School of Computing and Communications, Lancaster University, Lancaster, UK*

Correspondence should be addressed to Dingyi Fang; dyf@nwu.edu.cn and Zheng Wang; z.wang@lancaster.ac.uk

Received 20 September 2016; Accepted 21 November 2016; Published 17 January 2017

Academic Editor: Qingchen Zhang

Evil-Twin is becoming a common attack in smart home environments where an attacker can set up a fake AP to compromise the security of the connected devices. To identify the fake APs, The current approaches of detecting Evil-Twin attacks all rely on information such as SSIDs, the MAC address of the genuine AP, or network traffic patterns. However, such information can be faked by the attacker, often leading to low detection rates and weak protection. This paper presents a novel Evil-Twin attack detection method based on the received signal strength indicator (RSSI). Our approach considers the RSSI as a fingerprint of APs and uses the fingerprint of the genuine AP to identify fake ones. We provide two schemes to detect a fake AP in two different scenarios where the genuine AP can be located at either a single or multiple locations in the property, by exploiting the multipath effect of the Wi-Fi signal. As a departure from prior work, our approach does not rely on any professional measurement devices. Experimental results show that our approach can successfully detect 90% of the fake APs, at the cost of a one-off, modest connection delay.

1. Introduction

Smart homes consist of many intelligent, automation systems which are often connected to each other and the Internet through Wi-Fi to provide the inhabitants with sophisticated monitoring and control over the property's functions. Smart homes are increasingly becoming a target for cyber attackers [1–4]. Many of smart home targeting attacks exploit a technique called Evil-Twin where an adversary makes a rogue (i.e., Evil-Twin) access point (AP) with the same identity (or SSID) as an authorized AP, hoping that many of the wireless clients will connect to the rogue AP due to the commonly used automatic access point selection option [5]. An adversary can use an Evil-Twin AP as a platform to launch a variety of attacks, including privacy and data theft. Privacy concerns become evident because there are a large number of private data by various applications in the smart city, such as sensitive data of governments or proprietary information of enterprises [6].

How to detect Evil-Twin AP has recently received much attention [7, 8]. Generally speaking, there are two widely used approaches in this domain. The first approach uses traffic characteristics from the network flow [9, 10] to detect rogue APs. By analyzing information such as the packet arrival time, the request/response time of TCP ACKs, one can distinguish authorized APs from fake ones. Such approaches, however, depend on many environmental factors, such as the type and bandwidth of the network and traffic congestion (which can change from time to time). Therefore, such an approach is only applicable to a limited set of environments where the network traffic pattern is known ahead of time and is stable. The second approach, namely, fingerprint identification detection, uses hardware features [11–18], to identify rogue APs. This requires collecting fingerprint information from the hardware and systems software components (e.g., the firmware, the chip, and the driver) of the authentic APs. This approach is based on an assumption that it is difficult for the attacker to set up an AP with identical hardware information. However, building a fingerprint library is nontrivial and extracting the fingerprints from the APs could be time-consuming. These drawbacks make such approaches infeasible when real-time is an essential requirement.

This paper introduces a novel method for detecting Evil-Twin APs. Our approach targets smart homes. Our approach exploits the following observations: (1) the position of an AP is often fixed in a smart home environment; (2) the received signal strength indicator (RSSI) of a fixed AP is relatively stable. We consider the RSSI signal as the fingerprint of a genuine AP and use this information to identify rogue APs. One of the advantages of our approach is that we do not require any additional sensor/actuator infrastructure. Instead, we first use the stable RSSI to estimate the distance between the signal point and the receiving point [19–27] and then use the distance to detect rogue Evil-Twin APs. We show that our approach achieves on average a successful rate of over 90% with a one-off connection delay of less than 20 seconds.

The main contribution of this paper is a novel Evil-Twin attack detection system based on RSSI. We have shown that RSSI is a viable means for detecting rogue APs in smart home environments. Although our approach is evaluated in a smart home environment, similar ideas can be expanded to other Wi-Fi environments.

2. Background

SSID and BSSID are always used to identify Wi-Fi hot point since the protocol 802.11 does not define a strong sign to do it. In fact, both of them could be easily got by attacker, because the wireless network not only shares the media but also cannot control the signal range. Although the access point is protected by password and sophisticated encryption, for an experienced attacker, it is not difficult to crack it during a short time. The original 802.11 security organization that try to solve these problems was the Wired Equivalent Privacy (WEP). In spite of having mechanisms to provide authentication, confidentiality, and data integrity, WEP was found to be unsafe and trivially cracked after an attacker has gathered enough frames with the same initialization vector [28]. By actively accelerating the gather of frames, the latest WEP attack is able to complete breaking of WEP in under a minute [29]. WEP is increasingly being replaced by the Wi-Fi Protected Access (WPA). Nevertheless, to hold backward compatibility, WPA has not totally solved some security problems. Because control and management frames can be tricked and faked even with WPA enabled, wireless Local Area Networks (LANS) reserve impressionable to identity attacks and denial of service attacks [12]. Once the attacker got the password, they will soon forge the same one called the Evil-Twin AP (i.e., the rogue or fake AP), which is not easily recognized by user. Over the past few years, this kind of attack mainly exists in some public environments such as airports and cafes. However, as the development of the IoT, nowadays gigantic crowd-sourced data from mobile devices have become widely available in social networks [30], the attack value of private Wi-Fi rises rapidly, and the attack develops towards the private Wi-Fi in the smart home and other environments, such as privacy concerns that become evident on the cloud because there are a lot of private data in multimedia data sets [31]. Once the user connects the network to the fake AP, the intruder can control the network environment of the user, and further, privacy sniffing, malicious data tampering, and other advanced attacks can be realized. The behavior of the intelligent device even can be controlled, for instance, opening or closing an intelligent lock.

According to the IEEE 802.11 standard, when there are multiple APs nearby, the one with the strongest signal is to be chosen [16]. So the fake AP is always putting at the nearest of attack target in order to be chosen. This kind of attack can be called Fishing, which contains active Fishing and passive Fishing. Passive Fishing is named because the fake AP is just waiting for the connection from the terminal. This kind of attack cannot easily be found since it does not affect the Real AP; at the same time, the attack successful rate is not high. Active Fishing means that to connect with the terminal, fake AP cut the connection between Real AP and the terminal by Evil-Twin attack. Such attack can be carried out to precise attacks without affecting the other equipment except the target.

3. Attacking Scenarios

Attacking Scenarios. Figure 1 illustrates the scenarios where the Evil-Twin attack can be applied. Evil-Twin is designed to look like real Wi-Fi hotspots. In those scenarios, the adversary is able to set a fake AP to launch an Evil-Twin attack from a laptop. Its signal might be stronger to the victim than the Real AP. Once disconnected from the legitimate Real AP, the tool then forces offline computers and devices to automatically reconnect to the Evil-Twin, allowing the hacker to intercept all the traffic to that device. When people in smart homes are using the Internet through an Evil-Twin, they can unknowingly expose their passwords and other sensitive online data to hackers. According to the Wi-Fi Alliance, a sophisticated Evil-Twin can even control what websites appear when users access the Internet. That allows hackers to capture their passwords.

Our Assumptions. Our attacks require the adversary to set up the Evil-Twin at different locations. We believe that the adversary may not set the fake AP very close to the smart homes in order to avoid being caught. If a profile for the legitimate AP exists, the client device will automatically connect to the faked AP.

4. DRET Overview

Figure 2 is shown as the overview of DRET System. DRET is a system that helps wireless home owner to discover and prevent evil access points (AP) from attacking wireless users. The application can be run in regular intervals to protect your wireless network from Evil-Twin attacks. By configuring the tool you can get notifications sent to your alarm signal whenever an evil access point is discovered. Additionally you can configure DRET to perform DoS on the legitimate wireless users to prevent them from connecting to the discovered evil AP in order to give the administrator more time to react. However, notice that the DoS will only be performed for evil APs which have the same SSID but different BSSID (AP's MAC address) or running on a different

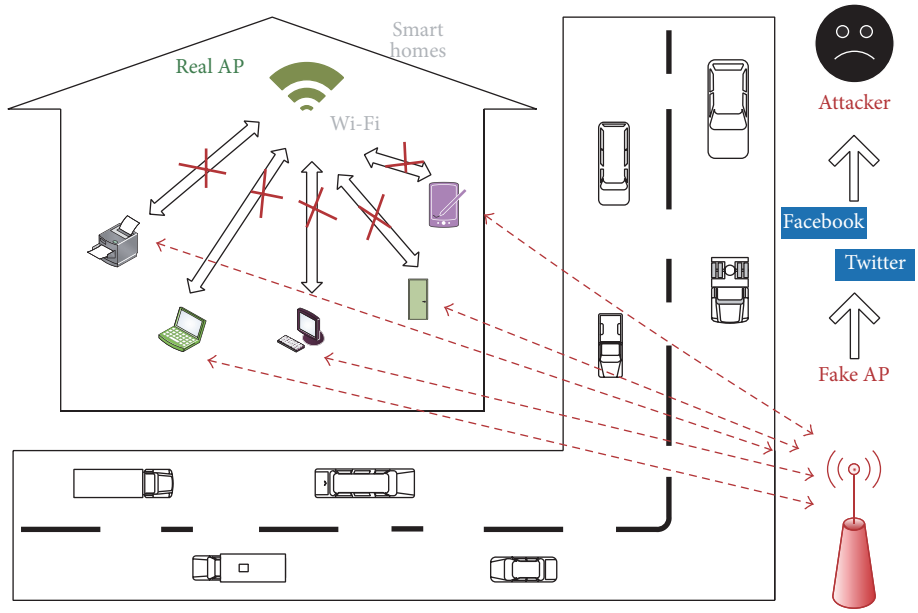


Figure 1: Example scenarios in which the attacker can easily launch an Evil-Twin attack to steal information using a fake AP. This kind of attack typically happens when a hacker constructs a mock (but still functional) Wi-Fi access point (AP) right at the place where there ought to be an original and legitimate access point. The reason this works so well is that, for a well-orchestrated attack, the illegitimate AP has stronger signals than the legitimate one and hence the unsuspecting users might log on to this mock-up connection and then use the Internet while sharing all their precious data, all the way from their user's IDs and passwords to credit/debit card information.

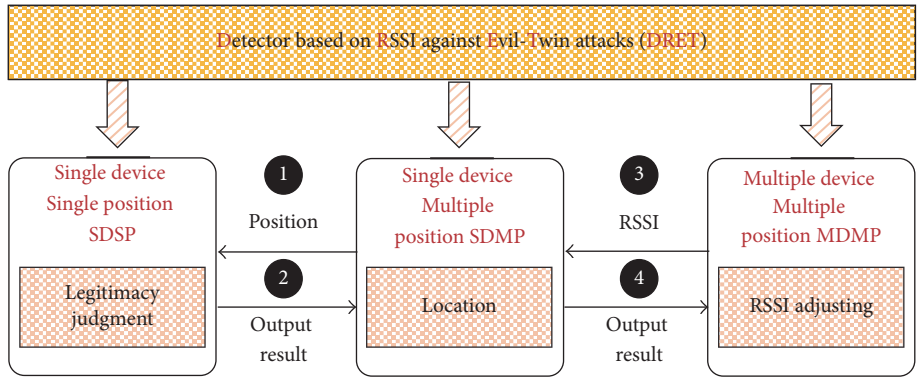


Figure 2: The overview of DRET System. DRET mainly consists of three parts (SDSP, SDMP, and MDMP).

channel. This method can prevent DoS from attacking your legitimate network.

Following a common practice in fake AP detection, DRET will choose different modules depending on different circumstances. SDSP meet the simple scenario such as during night and when nobody is at home. However, SDSP is limited and the success rate is closely related to the detector location. To address this limitation, SDMP is proposed, which locates the mobile phone firstly; the RSS fingerprint value is drawn to SDSP (❶), so the SDSP can determine the location of legitimacy (❷); the result returns to SDMP. Sometimes in many devices working in multiplaces, these devices need to use only one set of fingerprint data to check at the same time. MDMP will start; the RSSI is adjusted and then sent to SDMP (❸); the result done by SDMP returns to MDMP (❹).

5. Preliminaries

In order to construct a real environment, the attacker will do everything to improve the fake AP so that it has the same features of a Real AP, including traffic characteristics and hardware fingerprint characteristics. In real-world applications, the environment may have some negative effects on the identification of the target [32]. However, the attacker cannot forge the position of the Real AP. Recent literature advances Wi-Fi signals to "see" people's motions and locations. By detecting and analyzing signal reflection, they enable Wi-Fi to "see" target objects [33]. In smart homes, the intuition underlying our design is that each Real AP has its fixed position, and the attacker cannot put the fake AP exactly in the right place. Therefore, a new smart home fake AP detected method based on RSSI is proposed in this paper.

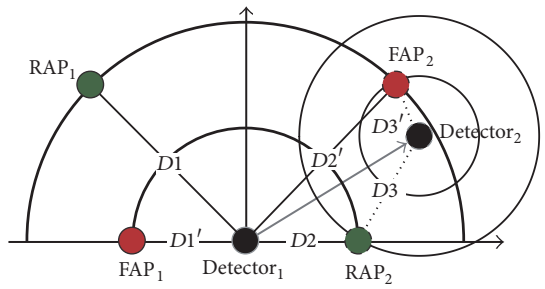


Figure 3: The figure shows two Real APs (in green) and two Fake APs (in red). The figure illustrates how the detector (in black) recognizes the FAP by using the differences of the RSSI that the APs locate differently.

Figure 3 is shown as the principle of fake AP detection based on RSSI. RAP and FAP are, respectively, represented Real AP and fake AP. Detector receives the signal from each AP. $D1$ is the distance between the $Detector_1$ and the Real AP, and $D1'$ is the distance between the $Detector_1$ and the fake AP. If $D1$ is greater than $D1'$, it means that the intensity of $Detector_1$ received from the fake AP is stronger than the real one. In general, when there exists multipath effect, detector always chooses the strongest signal in the homologous signals. So, undoubtedly, when the attacker turns on FAP_1, $Detector_1$ will choose it rather than the real RAP_1. But when the attacker turns off the FAP_1, $Detector_1$ will choose RAP. According to the upper analysis, we can easily identify the fake AP from the real one by comparing the RSSI of them. In this scene, If $RSSI_1'$ is greater than $RSSI_1$, it means that FAP_1 is fake AP.

However, there is another scene where the distance between the Real AP and detector is less than the fake ones. In this condition, no matter how open or shut down the fake AP is, the detector would always choose the Real AP. So, we should try to build a scene like the previous one, namely, moving the detection's position to $Derector_2$, making $D3'$ greater than $D3$; then we can detect the fake AP.

In free space, the path loss of signal propagation expresses signal attenuation, which is defined as the difference value between the effective radiated power and the received power. So the path loss in free space can be computed by the following formula. G_t and G_r separately express the antenna gain of the sender and the receiver. λ indicates the signal wave length; d is the distance between the sender and receiver.

$$\mathrm{PL\,(dB)} = 10\log\frac{P_t}{P_r} = -10\log\left[\frac{G_t G_r \lambda^2}{(4\pi)^2 d^2}\right]. \quad (1)$$

Frequency of Wi-Fi channel 1~13 is from $2.412 * 10^9$~$2.472 * 10^9$. And there exists $\lambda = c/f$ and $c \approx 3 * 10^8$ m/s, so the value range of λ is 0.1214~0.1244. We did some experiment to study factors effecting the attenuation and the attenuation curve is shown in Figure 4. In Figure 4(a), both of the sender and receiver have unity-gain, and the channel is 1. In Figure 4(b), both of the sender and receiver have unity-gain, and the channel is 13. In Figure 4(c), the antenna gain product of the sender and receiver is 100, and the channel is 13. From Figure 4, we can find the following rules. (1) From (a) and (b), we can find that the effect of channel on attenuation is very small. (2) From (b) and (c), we can find that antenna gain has a great influence on attenuation. (3) From (a), (b), and (c), we can find that the distance is the main factor to affect the attenuation, and the attenuation is less sensitive to the distance with the increase of distance.

RSSI (Signal Strength Indicator Received) is the intensity of the received signal; its value can be calculated by the following formula: RSSI = Transmit Power + antenna gain − path loss.

For a fixed transmitter and receiver, the Transmit Power and antenna gain are both constant, and the path loss is a function of the distance D, so RSSI can be expressed as $RSSI = f(d)$. Then d will be $d = f'(\mathrm{RSSI})$. Therefore, RSSI can be used directly to replace the distance for positioning.

In order to be simplify the calculation, we proposed signal space and signal distance. Signal distance can be abbreviated as sd; then $sd = |RSSI|$. In Figure 5, the left is the physical space, and the right is the signal space. Both of them take AP as the reference point. Points a, b, c, and d are the position of four mobile phones. In the physical space, the distance separately between a, c, and d is equal, less than the distance between b and AP. But there are obstacles at the points a and d, where the attenuation of the black obstacle is higher than the gray obstacle, so $sd_a > sd_d > sd_c$ and $sd_b > sd_c$. In general, the signal strength of straight line is the best when there is no obstacle, and wireless devices always give priority to the best signal when dealing with multipath effects. So, from the physical space to the signal space, the distance of their signal has some slight changes, which is shown as the right figure.

In order to verify that the RSSI can be used as the defection factor, we did an experiment. In normal circumstances, we build a fingerprint library by using the signal distance. Terminal MX3 is used as director to collect RSSI signal and the TL-WR882N is used as AP. The distance between them is 5 m, and data collection rate is 2 times per second. We collected about 14000 of the total data, keeping surrounding environment not changed during the process of collecting data, except when someone walked across. Its probability distribution histogram is shown in Figure 6.

By analyzing the experiment data, it is found that the measured value of the actual measurement is near to a stable value, and the probability distribution is approximately normal distribution. That means the RSSI can be used as the defection factor.

Actually, it seems that both of the fake and Real AP is similar to the detector, which are difficult to be distinguished. According to multipath effect, the detector will select the one with the strongest signal to associate and compute the distance between the selected AP and it, which will be compared with the distance recorded in signal distance fingerprint database. If they are different, that means the AP should be forged. The mobile phone will be used as the detector. Depending on whether the mobile phone used as a detector in smart home is moving or not, two different kinds of solution have been proposed in this paper; they are a single fixed position detection and the multiposition collaborative detection.

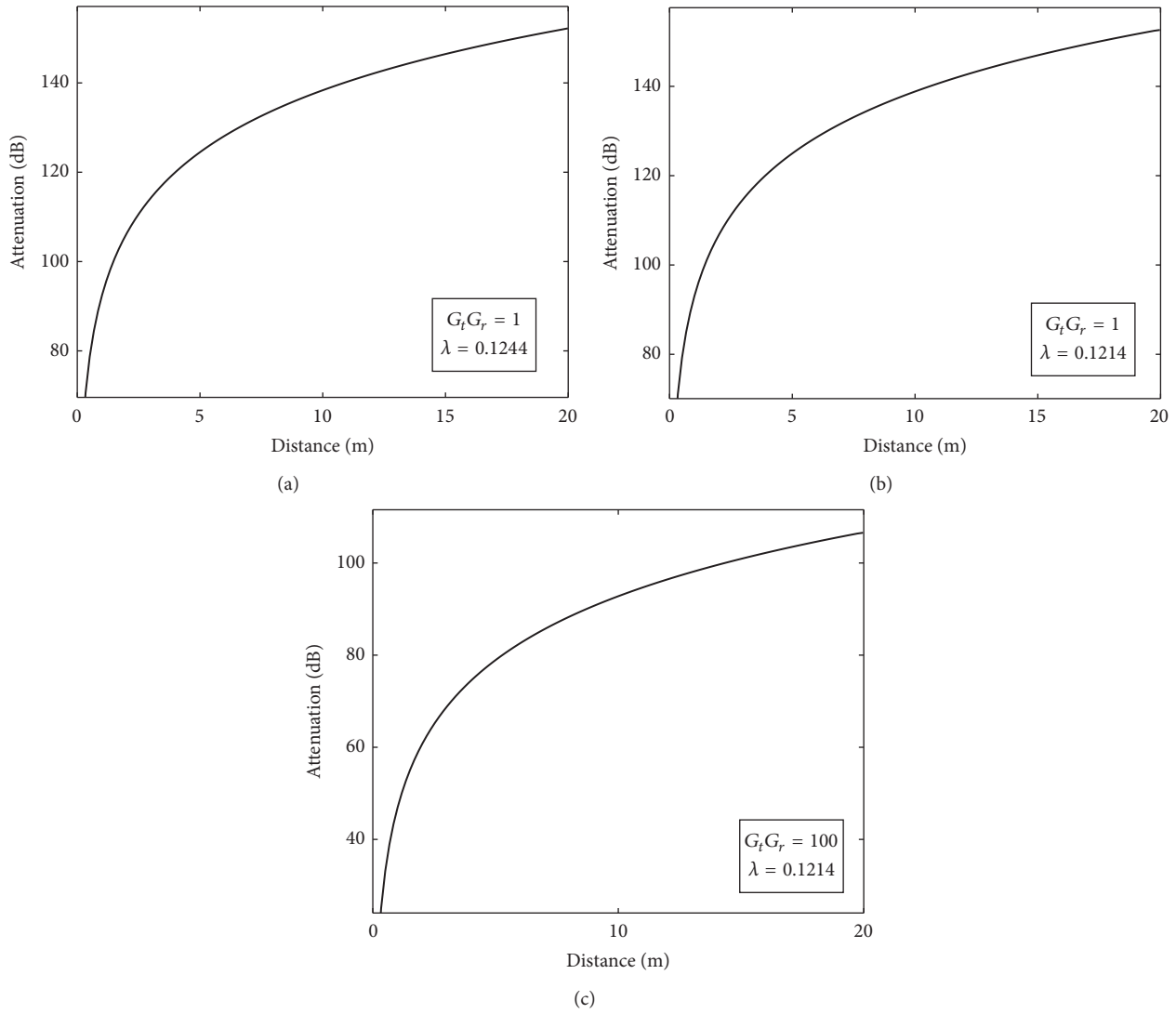


Figure 4: Signal attenuation curve.

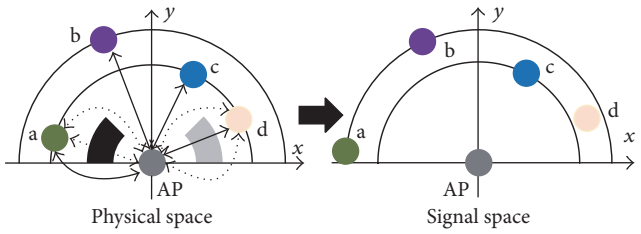


Figure 5: Physical space convert to signal space.

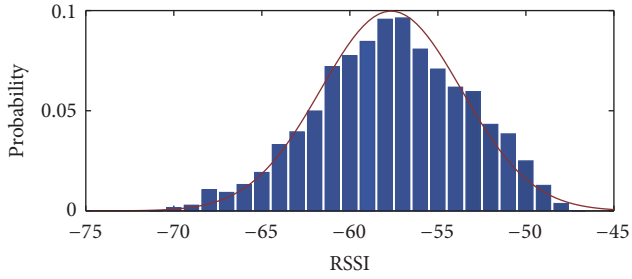


Figure 6: Probability distribution histogram.

6. Automated Detection Analysis

6.1. A Single Fixed Position Detection. Smart homes devices still need work under networking even if there is nobody at home, so the detector can also finish the detecting of false AP. Therefore, we install the detector in a fixed position, and let it work 24 hours. Detector establishes target AP RSSI fingerprint library in normal sense, which would be used as sample when detecting. Only the detected distance is within the error range of distances recorded in fingerprint database; it is considered as the fake AP; otherwise, it is true AP.

It is assumed that the deployment of hot spot and detector is shown in Figure 7. The position of fake AP and true AP is different, but the other features are the same, such as network

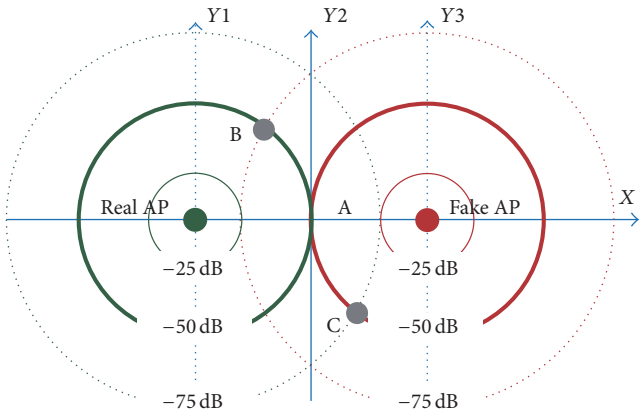


Figure 7: A single fixed position detection.

Table 1: FSSI and variance in the security state.

Location	Average	Variance
A	$\mu_A = -50$	σ_A
B	$\mu_B = -50$	σ_B
C	$\mu_C = -75$	σ_C

card hardware features, antenna gain, and stability. A, B, and C are the positions of three detectors. The signal intensity of true AP and fake AP is the same in the position A (shown as $Y2$ axis). The signal intensity of true AP is stronger than ones of fake AP in the position B and the opposite in position C.

In the security state, that is, where the fake AP does not exist, the RSSI and variance of signal intensity separately received by three detectors at positions A, B, and C are shown in Table 1.

Fake APs working will lead to multipath effect. Therefore, it is assumed that P_A, P_B, and P_C are the probability of selecting true AP signal in A, B, and C. Under ideal conditions, $0 \le P_C < P_A = 0.5 < P_B \le 1$, and the new average and variance are shown in Table 2. Both of them wave in a certain range of fluctuation due to kinds of factors like the multipath effect, the external interference, and so forth. It is assumed that the average an variance meet the following conditions: $\mu - M \le \mu \le \mu + M$, $\sigma \le \Sigma$.

From Figure 7, we can see that when the detector is in region C, it will select fake AP whose signal intensity is stronger than the Real APs, which can be described with a formula like $\mu' > \mu$. When $\mu' > \mu + M$, we can say that there exists a fake AP in the network. When the detector is in region A, $\mu' = \mu$; that means we cannot distinguish the Real AP and the fake one. In region B, although the signal intensity of Real AP is higher than fake AP, but the detector considers both of them are the same signal; the latter still cannot be detected.

As analysis shows detector and Real AP cannot be too close that will lead to high misdetection rate, so the best deployment location of detector is in region C where the signal is weak, far away from the Real AP and near the fake AP.

Table 2: FSSI and variance when fake AP is working.

Location	Average	Variance
A	$\mu'_A = \mu_A = -50$	$\sigma'_A = \sigma_A$
B	$-75 < \mu'_B < -50$	$\sigma'_B > \sigma_B$
C	$75 < \mu'_C < -50$	$\sigma'_C > \sigma_C$

6.2. Multiposition Detection. Obviously, a single fixed position detection method can only solve part of the problem. In this part, multiposition detection is proposed. Multiposition detection relies on mobile phones; with it we can convert multiposition to single fixed position detection. So, first what we need to do is determine the position of the mobile phone. The most well-known and highly accurate positioning method is GPS, while GPS devices have been known to not work very well indoors. In this paper, we use the Wi-Fi signal for locating the position of mobile phone by three-point positioning method. With the popularity of Wi-Fi, there are almost always more than three Wi-Fi hotspots that will be found when we are indoors.

As shown in Figure 8, AP_1, AP_2, and AP_3 are three different APs, assuming their positions are known. O is the mobile phone's position. The original distance can be defined as sd which represents the distance between AP and mobile phone. $sd_i = |OO_i|$, $i = 1, 2, 3, 4, 5$. So AP_1, AP_2, and AP_3 can locate the position of the mobile phone in the signal space. Then we can convert the multiposition detection to a single fixed position detection.

There are two stages in multiposition cooperative detection: fingerprint gathering stage and detection stage. The first stage should be done in a safe state; we collect the RSSI information both of reference AP and target AP in many different positions, to build a fingerprint library. In the detection stage, using reference AP to locate the phone and the fingerprint data in a single fixed position detection, the program framework is shown in Figure 8; we can locate the mobile phone's position by using reference AP and then using the method mentioned in the previous chapter to detect.

In Figure 9, AP_0 is the target AP, AP_2~AP_n are the candidate's reference AP, and the whole process can be divided into the following 5 steps:

Step ❶: RSSI acquisition.

Step ❷: effective data selection.

Step ❸: establishment of fingerprint database.

Step ❹: mobile position determination.

Step ❺: validity judgment.

6.2.1. RSSI Acquisition. In the experiment, the value of RSSI is collected by mobile phone; the detection program can import corresponding management package and call relevant interface (Android: android.net.wifi.*; IOS: SystemConfiguration/CaptiveNetwork.h) so that it can make mobile phone acquire enough RISS value in daily routines.

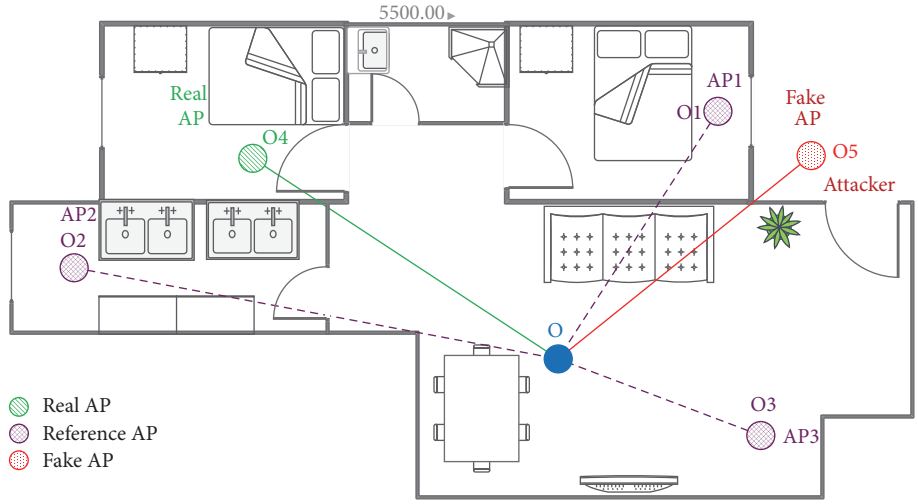


Figure 8: Multiposition detection transformation. The figure shows that any three APs could be chosen as reference in the signal space. They are used to locate the positions of the mobile phone which is a detector in smart homes.

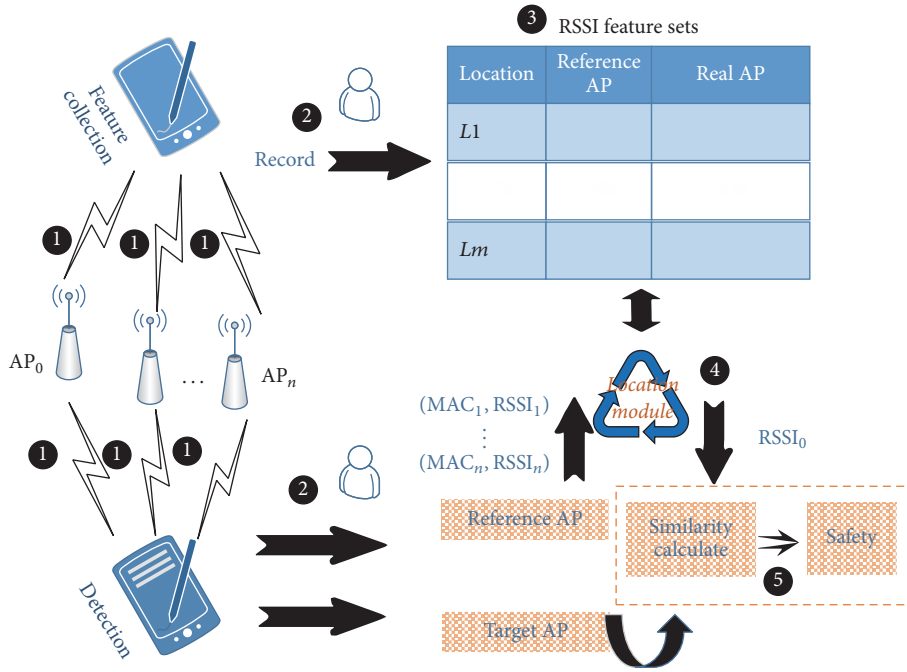


Figure 9: Multiposition detection framework.

6.2.2. Effective Data Selection

Effective RSSI Values Selection. It is a challenging job to choose the right RSSI values since the mobile phones are always moving. However the RSSI value we need should be waved in a small range, which is shown in Figure 10. The data in two boxes are what we want; the others are generated by mobile phone when it is moving. When the distance between mobile phone and AP is 1 m and there is no interference, it can generate the data in the first box. Data in the second box is generated in the condition that the distance between mobile phone and AP is 4 m and there are two sources of interference. The other data is generated in the condition that someone takes the mobile phone and go around the house with the speed of 1.5 m/s.

In the first experiment, variance increment method is used to judge whether the mobile phone is moving. It is assumed that the size of sliding window is 120. When the amount of data is less than the window, it is invalid data.

$$W_i = \{r_{i-\text{ws}+1}, r_{i-\text{ws}+2}, \ldots, r_{i-1}, r_i\} \quad i \geq \text{ws},\ r_i \in R. \tag{2}$$

R is the whole RSSI sequence, r_i is the value of RSSI, and ws is the window size.

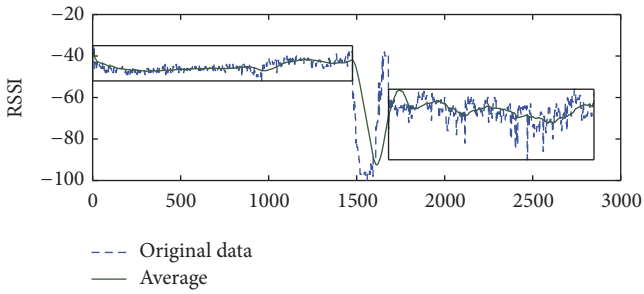


Figure 10: The RSSI sequence.

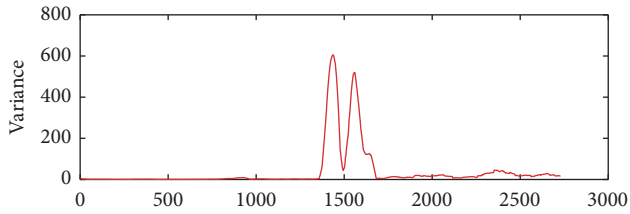


Figure 11: The RSSI sequence variance.

The variance can be used to measure the deviation between the RSSI data and the mean value of the window. The variance of W_i is σ_i which expresses the data fluctuation of W_i. The greater the data fluctuation, the greater the variance.

As shown in Figure 11, the window size is 120, with two peaks in the middle corresponding to the moving process; that is, it corresponds to the parts not in those two boxes in Figure 10. However, the cause of the big variance is not necessarily a person's movement; the stability of the signal will also affect it. Therefore, the slope of the variance curve is used to determine whether the current is moving. The variance increment

$$k(i) = \frac{d_{\sigma_i}}{d_i} = \frac{\sigma_i - \sigma_{i-1}}{i - (i-1)} = \sigma_i - \sigma_{i-1}. \quad (3)$$

In Formula (3), σ_i is the variance of W_i and σ_{i-1} is the variance of W_{i-1}.

The improved results are shown in Figure 12. When $k(i)$ is near to 0, it means that the original variance is stable in a certain range; that also means the mobile phone is not moving or moving in a small range. We set a threshold to detect whether the mobile phone is moving. If $|k(i)| \le K$, the mobile phone is considered to be stable; otherwise it means the position of mobile phone has changed.

Those sequences with a stable position have the following characteristics:

Start point: $[|k(i)| \le K] - \mathrm{ws}(+1)$.

End point: $[|k(i)| > K] - \mathrm{ws}/2$.

Effective Reference AP Selection. In order to improve the accuracy of multiposition detection, it is needed to improve the accuracy of the location. Because of the complexity of the wireless signal transmission in the indoor environment,

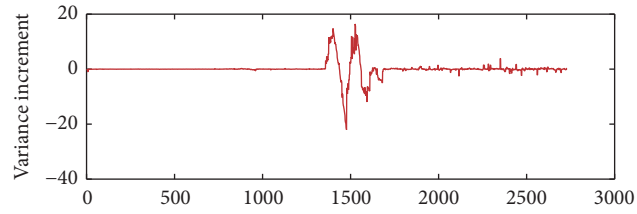


Figure 12: RSSI sequence variance.

the AP signal is not stable. In the network environment, a position can be detected by more than one AP. Therefore, signal stability and the relevance with target AP are the two factors in choosing AP. Relevance here means that both the target AP and the reference AP moving with the mobile phone; that is why the fluctuations of the variance between the target AP and the reference AP should be consistent.

We use dynamic (dynamic time warping, DTW [34]) algorithm to calculate the distance and determine the validity of the reference AP. DTW is a method that calculates an optimal match between two given sequences (e.g., time series) with certain restrictions. The sequences are "warped" nonlinearly in the time dimension to determine a measure of their similarity independent of certain nonlinear variations in the time dimension. This sequence alignment method is often used in time series classification.

As is shown in Figure 13, (a) calculates distance without using dynamic time but (b) uses it; by using dynamic time, (b) can reach the minimum distortion when it comes to calculate the distance.

When selecting the effective reference AP, each AP is considered as the candidate reference AP. The large number of its variance increment is stored as well as the distance between its variance increment sequence and the target's. After getting the distance of all candidate reference APs and target APs, all candidate reference APs will be ordered by the distance. The smaller the distance, the better the effectiveness. Therefore, four candidate reference APs with the minimum distance will be chosen as the reference APs to locate the mobile phone's position. In general, three points are enough for location. In order to prevent that one of the three reference APs from failure, so we choose four reference APs from the candidate lists.

6.2.3. Establishment of Fingerprint Database. The RSSI fingerprint library (RSSI-MAP) is built by the RSSI sequence generated in previous section. RSSI-MAP is shown in Table 3. $R_J = (r_{1,J}, r_{2,J}, \ldots, r_{L,J})$ represent the fingerprint information in RSSI-MAP. J is the position where the mobile phone is stayed for detecting. L is the number of candidate reference APs. r is the fingerprint information of AP, which can be described by triple like $r(\overline{\mathrm{rssi}}, \mathrm{var}, \mathrm{len})$. Items in triple represent the average, variance, and length of RSSI sequence.

6.2.4. Mobile Position Determination. $R_T = (r_{1,T}, r_{2,T}, \ldots, r_{L,T})$ represents RSSI fingerprinting information of the reference APs detected at the position T. $R'_T = r'_{0,T}$ represents the

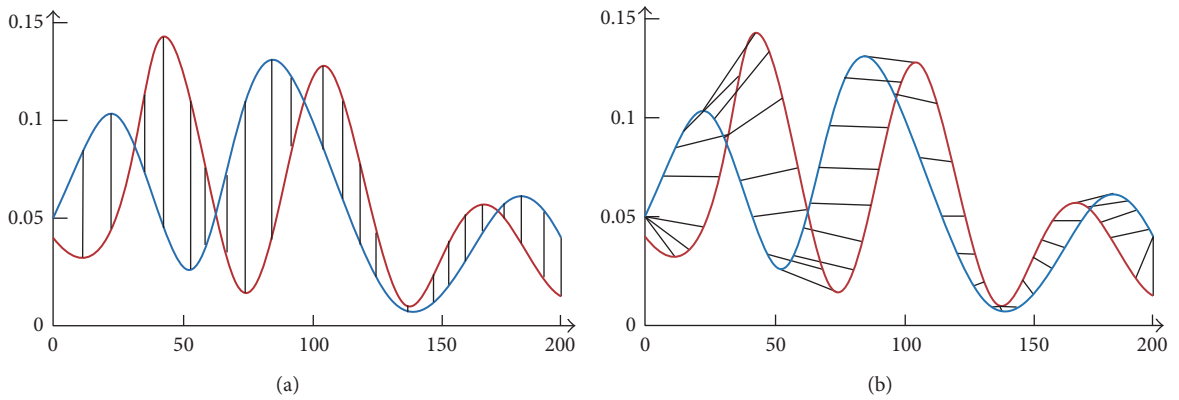


Figure 13: Dynamic time warp (DTW).

Table 3: Structure of RSSI-MAP.

Location	Reference AP	Target AP
1	$R_1 = (r_{1,1}, r_{2,1}, \ldots, r_{L,1})$	$R'_1 = r_{0,1}$
2	$R_2 = (r_{1,2}, r_{2,2}, \ldots, r_{L,2})$	$R'_2 = r_{0,2}$
⋮	⋮	⋮
J	$R_J = (r_{1,J}, r_{2,J}, \ldots, r_{L,J})$	$R'_J = r_{0,J}$

RSSI fingerprinting information of the target AP detected by the position T. $\mathrm{Dist}(R_T, R_J)$ is the distance between R_T and R_J. $\overline{\mathrm{rssi}_{i,T}}$ is the average value of RSSI for reference AP; $\overline{\mathrm{rssi}_{i,J}}$ is the average of the RSSI sequence for reference AP. J is the position where the distance between T and one in RSSI-MAP is the shortest. When there are more than three reference APs, we can locate the mobile phone.

$$\mathrm{Dist}\left(R_T, R_J\right) = \sqrt{\sum_{i=1}^{L} \left(\overline{\mathrm{rssi}_{i,T}} - \overline{\mathrm{rssi}_{i,J}}\right)^2}. \tag{4}$$

$\mathrm{Dist}(R_T, R_J)$ in Formula (4) depend on the number of L, in order to reduce the effect on Dist_T that the number of reference AP is different in different position. The formula is improved as the following.

$$\mathrm{Dist}_T = \min\left[\frac{\mathrm{Dist}\left(R_T, R_J\right)}{L}\right]. \tag{5}$$

When L is greater than or equal to 3, the fingerprint of the first three APs can be used for location by using Formulas (4) and (5). When L is equal to 2, there will be more than one position and all of them have the same distance. Then we should choose the one who is the nearest one with the target AP. When L is equal to 1, in order to increase the accuracy of the positioning, the variance is used to measuring the similarity between position T and position J. From the previous section, the RSSI form one AP at the same position which is approximate normal distribution; that is, the RSSI sequence is represented as follows:

$$P\left(\mathrm{rssi}\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{(\mathrm{rssi}-\mu)^2/2\sigma^2}. \tag{6}$$

In Formula (6), $\sigma = \mathrm{var}$; $\mu = \overline{\mathrm{rssi}}$.

In the information theory, KL [35, 36] (Kullback-Leibler, divergence) can be used to describe the difference between two probability distributions of Q and P; $D_{\mathrm{KL}}(P \parallel Q)$ is the information loss caused by that Q which is used to fit the true distribution P. So the distance between the T and the RSSI probability distribution can be calculated using the KL divergence. KL divergence is defined in

$$D_{\mathrm{KL}}\left(P \parallel Q\right) = \sum P\left(i\right) \ln \frac{P\left(i\right)}{Q\left(i\right)}. \tag{7}$$

So, we can get formula (8) from formula (6) and formula (7).

$$\begin{aligned}\mathrm{Dist}\left(R_T, R_J\right) &= D_{\mathrm{KL}}\left(R_T \parallel R_J\right) \\ &= \sum_{\mathrm{rssi}=-100}^{0} \frac{P\left(\mathrm{rssi}\right)}{2}\left[\frac{\left(\mathrm{rssi}-\mu_1\right)^2}{\sigma_1^2} - \frac{\left(\mathrm{rssi}-\mu_2\right)^2}{\sigma_2^2}\right].\end{aligned} \tag{8}$$

In the formula (8),

$$\begin{aligned}\sigma_1 &= \mathrm{var}_{L,T}, \\ \mu_1 &= \overline{\mathrm{rssi}_{L,T}}, \\ \sigma_2 &= \mathrm{var}_{L,J}, \\ \mu_2 &= \overline{\mathrm{rssi}_{L,J}}, \\ P\left(\mathrm{rssi}\right) &= \frac{1}{\sqrt{2\pi}\sigma_1} e^{(\mathrm{rssi}-\mu_1)^2/2\sigma_1^2}.\end{aligned} \tag{9}$$

Then, according to the distance got by formula (8), the nearest neighbor algorithm is used to find the corresponding position in the J RSSI-MAP.

6.2.5. Legitimacy Judgment. $\max(\overline{\mathrm{rssi}})$ represents the maximum mean of target RSSI at position J. It can be easily query in RSSI-MAP when we find the position J. $\overline{\mathrm{rssi}}$ is the mean value being detected. Then, there is $\mathrm{Diff}_T = \overline{\mathrm{rssi}} - \max(\overline{\mathrm{rssi}})$.

If $\mathrm{Dist}_T \le M$ and $\mathrm{Diff}_T \le 0$, it is safe and there is no fake AP.

If $\mathrm{Dist}_T \le M$ and $\mathrm{Diff}_T > 0$, it is unsafe and there exits fake AP.

If $\mathrm{Dist}_T > M$, fingerprint database should be updated. You can find the details in next section.

6.2.6. Dynamic Update of Fingerprint Database. The dynamic update of RSSI fingerprint database consists of two parts: one is the addition of the new fingerprint, and the other is the update of the existing fingerprint.

The new fingerprint should be added because of various reasons in the training phase of the RSSI fingerprint database. It cannot cover all the spatial subregions of M, so it is necessary to improve the fingerprint database in the later stage.

The update of the existing fingerprint is caused by environmental changes such as survival status of reference AP, the correlation between the candidate reference AP and the target AP, and the change of the reference AP's position. At this point, we need to update the fingerprint information which already exists in the fingerprint database in detection stage.

$$\left[R_J\left(r_{1,J}, r_{2,J}, \ldots, r_{L,J}\right), R'_J\left(r_{0,J}\right)\right]. \tag{10}$$

Assume there are four valid candidate reference APs; they are $\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_3, \mathrm{AP}_4$, and the relationship or their effectiveness is as the following: $E1 > E2 > E3 > E4$; then there is $\mathrm{Dist}_T = \mathrm{Dist}_T(\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_3)$. The corresponding position is J.

When there is $\mathrm{Dist}_T > M$

$$\begin{aligned}
\mathrm{Dist}_{T3} &= \mathrm{Dist}_T\left(\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_4\right),\\
\mathrm{Dist}_{T2} &= \mathrm{Dist}_T\left(\mathrm{AP}_1, \mathrm{AP}_3, \mathrm{AP}_4\right),\\
\mathrm{Dist}_{T1} &= \mathrm{Dist}_T\left(\mathrm{AP}_2, \mathrm{AP}_3, \mathrm{AP}_4\right).
\end{aligned} \tag{11}$$

If $\mathrm{Dist}_{Ti} \le M$, then we can use $r_{i,T}$ instead of $r_{i,J}$ in the RSSI-MAP to update the existing fingerprint. If $\mathrm{Dist}_{Ti} > M$, then put (R_T, R'_T) into the RSSI-MAP. If $\mathrm{Dist}_{Ti} \le M$ and $r_{i,t}\mathrm{len} \ge r_{i,j}\mathrm{len}$, then we can use $r_{i,T}$ instead $r_{i,J}$ in the RSSI-MAP.

7. Evaluation in SPD and MPD

In order to verify the feasibility and effectiveness of the AP Evil-Twin detection method based on RSSI, we implement a number of experiments.

We use the Terminal MX3 to collect RSSI signal. The TL-WR882N is used as the true AP. A fake AP has been simulated by hostapd in a notebook. The experiment is done in a room with 100 square meters. In the detection phase, we set the different $F - R$ ($F - R$ is defined as the mean difference, resp., between the fake AP and the true AP's RSSI. The mean difference is equal to the distance between two APs.).

7.1. Experiment and Assessment for Single Position Detection

Discussion of Sliding Window Size. The previous section shows the size of the sliding window affects the delay rate and false negative rate of detection. That means the bigger the window, the higher the delay rate, and the higher the false negative rate. In order to find a suitable value for the size of sliding window, we design an experiment like the following.

In order to verify the effect of window size on the delay, we set the mean difference, respectively, between the fake AP and the true RSSI as 25 and 10; that is, $F - R = 25$ and $F - R = 10$. The window size in turn is 1, 40, 80, 120, 160, 200, and 240. The safety threshold value for each round of detection is the maximum mean of RSSI in 30 minutes. There are 14 sets of experiment; each set of experiment will be done 30 times, and the result is as shown in Figure 14. From (a) we can see that when the difference of mean between true AP and fake AP is bigger, the delay rate is smaller. When the window size is 120, the average delay time is less than 20 s.

To verify the effect of window size on accuracy, when it is in the condition that $F - R = 10$, we set the windows size in turn: 1, 40, 80, 120, 160, 200, and 240. After the test program running 10 minutes, open the fake AP and let it run for 3 minutes then close it for 3 minutes, because it needs a certain delay that the mean value is changed from abnormal status to normal status.

The mean from abnormal status returning to normal needs a certain delay, so if there occurs wrong or missed detection in every 3 minutes after the delay time, it will be assumed as a wrong one. If there is wrong or missed detecting after delayed time, it is considered as the error status. This experiment is done 50 times, and the result is shown on the right in Figure 14. According to the experiment results, when the window size is 80, 120, and 160, the accuracy is more than 98%. If the windows size is too small or too big, the accuracy is lower since the false positive rate is higher.

Discussion of Threshold Value. In this experiment, we set the window size as 120 and the $F - R$ as 25 or 10. Assume that the threshold value is Rmax, Rmax $-$ 2, Rmax + 2, Rmax + 4, and Rmax + 8. So there are 10 sets of experiment. In each experiment the following step is done 50 times. After the test program running 10 minutes, open the fake AP and let it run for 3 minutes and then close it for 3 minutes. We can get the result of this experiment from Figure 15, when the security threshold value is Rmax and the accuracy is up to 96%. When the security threshold value is Rmax + 2, the accuracy of the condition is $F - R = 25$ up to 100% and $F - R = 25$ is 99%.

Discussion of Distance. In this experiment, we set $F - R =$ 0, 5, 10, 15, and 20, and the threshold value is Rmax. Each experiment is to be done as the following step 50 times. After the test program is running for 10 minutes, open the fake AP and let it run for 3 minutes and then close it for 3 minutes. We can get the result of this experiment from Figure 16. When $F - R = 10$, the accuracy is more than 96%; the missing rate is less than 3%.

7.2. Experiment and Evaluation of Multiposition Cooperative Detection

Validation of Variance Increment Method. In this experiment, the window size is 120, and K is 4; then split the RSSI sequence using Variance increment method. The result is shown in Table 4. Dropping out the fragment whose length is shorter than 120, then we can get two effective RSSI sequence fragments (S_1 and S_10), the total length is 2598,

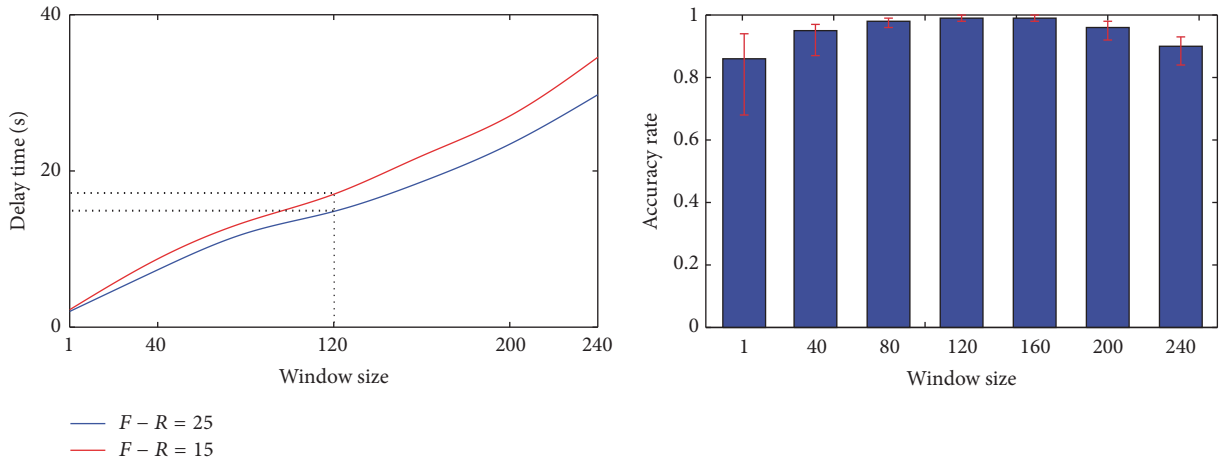


Figure 14: Effect of window size on delay and accuracy.

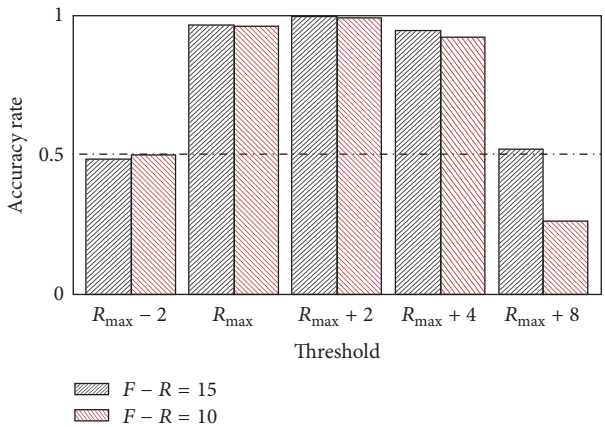


Figure 15: Effect of safety threshold on the accuracy of detection.

Table 4: First time to split the RSSI sequence.

Flag	Range	Length	Range	Mean
S_1	1–1422	1422	[−52, −35]	−45.15
S_2	1366–1431	66	[−44, −39]	−42.5
S_3	1424–1502	79	[−84, −38]	−50.04
S_4	1489–1560	72	[−100, −64]	−91.17
S_5	1507–1569	63	[−100, −87]	−95.95
S_6	1552–1620	69	[−100, −72]	−90.91
S_7	1609–1718	110	[−76, −38]	−56.54
S_8	1660–1726	67	[−75, −40]	−56.68
S_9	1669–1731	63	[−75, −40]	−59.95
S_10	1861–2848	1168	[−90, −56]	−66.37

and the effective fragment length was 2605 in the original data sequence. So the accuracy is 99.7%.

The Validity of DTW Algorithm. To verify that the DTW algorithm could be used to choose the valid AP, we open the detecting software which could find all the AP and get their RSSI. Then we let the detecting software move with the speed of 1.5 m, staying at three different locations and staying at each place for 15 minutes. At the end, there are 28 APs being found, including 1 target AP and 27 candidate reference APs. For each of 27 candidate reference APs, we use DTW algorithm to calculate the distance of variance increment sequence between target AP and it. Finally, we are successful to find four suitable reference APs.

The Validity of Localization Algorithm. In a room with 100 square meters, we collect a set of data per 4 square meters. So there are 25 sets of data. In detecting stage, we stayed at every position for 5 minutes, then moving to another position with the speed of 1.5 m/s. For the four suitable reference AP found in previous section, there are three kinds of conditions; that is, the first 4 AP should be considered as the reference AP, and the first 3 and the first 2, respectively, calculate their Euclidean distance. When there is only one reference AP, the accuracy of location is 62%. When there are two reference APs, the accuracy of location is 85%. When there are three reference APs, the accuracy of location is 90%.

The Validity of Multiposition Cooperative Detection. We play a role of an attack, simulating a fake AP in a notebook. And the experiment is done still in a room with 100 square meters, dividing it into 25 regions. In each region, we collect data for every 30 minutes and use the maximum mean of this region as the safety threshold. In detecting stage, we stayed at every position for 5 minutes, then moving to another position with the speed of 1.5 m/s. Experiments were carried out for 200 times, 100 times is to open the fake AP, and the other 100 times is to turn off the fake AP. When the fake AP is turned on, if there is any position detected by the fake AP, then the detection is successful, if all the positions are not detected by the fake AP, then the detection fails. Close the fake AP; if there is any position to detect the false AP, then the detection fails; if all the positions are not detected in the fake AP, then the

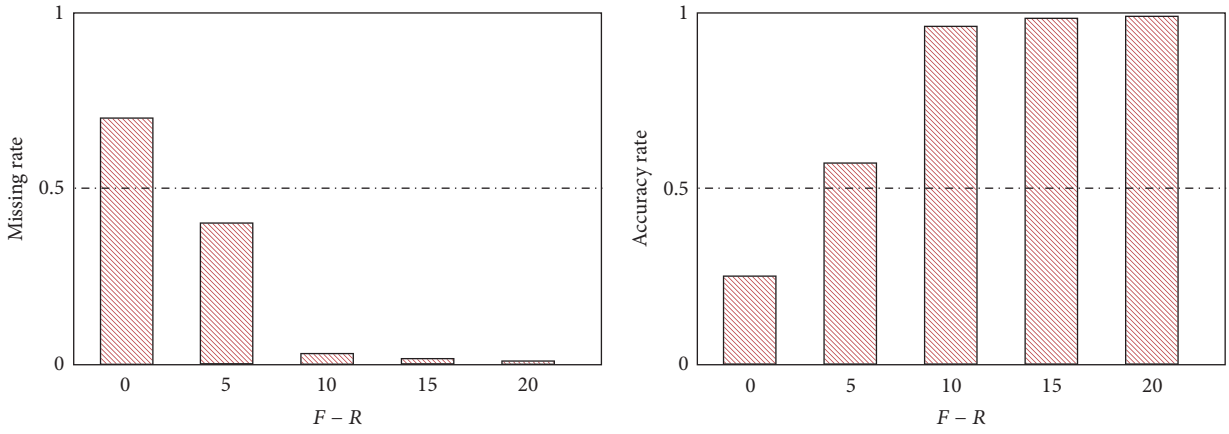


Figure 16: Effect of distance on the detection results.

detection is successful. When there is only one reference AP, the accuracy of location is 58%. When there are two reference APs, the accuracy of location is 80%. When there are three reference APs, the accuracy of location is 90%.

8. Related Work

At present, most Evil-Twin detection methods work for the public Wi-Fi environment. There are two key approaches in this domain. One is based on hardware feature; the other is flow feature.

The hardware feature testing method utilizes the characteristic that different network card chips and different drives possess different fingerprint features to set up a fingerprint feature library and decide whether the fake AP existed or not through matching fingerprint data in the fingerprint feature library during testing. Bratus et al. [9] send some SIMULATING frames which possess false formats but are not prohibited by a standard protocol. Although different network card chips or drives have different responses to various SIMULATING frames, the testing method is easy to be found by an intruder. McCoy et al. [11] characterize the drivers during the "active scanning period." This method is undefined in the IEEE 802.11 standard on the frequency and order of sending probe requests. Therefore, each manufacturer employs its own algorithm. This technique cannot distinguish between two devices using the same network card and driver. So this technique may not be used for identifying individual devices. However, the attacker cannot forge the position of the Real AP. In smart homes, the intuition underlying our design is that each Real AP has its fixed position, and the attacker cannot put the fake AP exactly in the right place. Desmond et al. [12] used fingerprint client station, which sends probe requests in light of periodic characteristic by surveying probe requests. The period itself is attached to slight variations. Far from being consistent, these variations can be clustered. With enough detection time, each cluster slowly derives, with a slope proportional to the time skew. This work is able to particularly identify client station; however, this requires more than one hour of traffic and is only applicable to client stations. In a word, McCoy et al. [11] and Desmond et al. [12] utilize the characteristic that different wireless network cards send different probe request frames with different periods during scanning to set up the fingerprint library. As the equipment only sends a small number of probe request during joining the network and the method can be valid when passive scanning is used, the expensive time overhead and the relatively bad real-time property are involved; Neumann et al. [13] utilize the arrival time of interframe space to identify the wireless equipment, but the characteristic can be faked by the intruder and the testing method based on the characteristic can be bypassed. The testing methods for the hardware fingerprint feature of the equipment above-mentioned cut both ways: various fake AP can be tested effectively and the cost of faking the hardware feature of the intruder is relatively high; the fingerprint database can be built in many ways [37], but the cost of building the hardware feature fingerprint library is high, the time for extracting the hardware fingerprint is long, the testing real-time property is worse, and the expansibility is bad. However, Our approach builds the feature fingerprint library without collecting deliberately. You will achieve the feature fingerprint library as soon as you open the phone.

According to the flow feature testing method, the network flow feature is different when the fake AP is existent or nonexistent; so, whether Evil-Twin AP is existent or not can be tested. The method is excellent in extendibility but also has some disadvantages. Beyah et al. [14] utilize the arrival time space of each data packet to build a flow feature library; as the method is influenced by flow shaping greatly, the practical operation and the applicability are not good; Wei et al. [15] propose that the arrival time of the ACK data packet in a TCP protocol can be used to set up the flow feature library; as the arrival time is influenced by TCP, the testing efficiency is limited; Sheng et al. [16–18] propose that data round trip time can be used to test whether the fake AP is existent or not, but the data round trip time is influenced by the network type, the bandwidth, and the state of congestion at the same time.

Besides, Han et al. [38] put forward the wireless fake AP attack in an in-vehicle network and, meanwhile, give the testing method based on RSSI. The method requires that all

of the APs are equipped with GPS modules to report their own positions; a user judges whether the fake AP is existent or not through whether the measured RSSI is matched with the position or not. The method can effectively test the fake AP attack in the in-vehicle network but is not suitable for indoor environment because the GPS signal is weakened, even shielded, indoors.

9. Conclusions

This paper has presented a novel approach to detect fake APs in a smart home environment. Our approach uses RSSI as the fingerprint of the authentic AP to detect fake APs. We have proposed two methods to identity fake APs in two different scenarios where the genuine AP locates on a single, fixed, or multiple positions. Our experimental results show that our approach can detect 90% of the fake APs with little extra overhead to the communication delay time.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grant Agreement no. 61672427, no. 61672428, and no. 61272461, the Key Project of Chinese Ministry of Education under Grant Agreement no. 211181, the International Cooperation Foundation of Shaanxi Province, China, under Grant Agreement no. 2013KW01-02 and no. 2015 KW-003, the Research Project of Shaanxi Province Department of Education under Grant no. 15JK1734, the Research Project of NWU, China (no. 14NW28), and the UK Engineering and Physical Sciences Research Council (EPSRC) under Grants EP/M01567X/1 (SANDeRs) and EP/M015793/1 (DIVIDEND).

References

[1] M. Chan, D. Estève, C. Escriba, and E. Campo, "A review of smart homes—present state and future challenges," *Computer Methods and Programs in Biomedicine*, vol. 91, no. 1, pp. 55–81, 2008.

[2] J. Schulz-Zander, L. Suresh, N. Sarrar, A. Feldmann, T. Hühn, and R. Merz, "Programmatic orchestration of wifi networks," in *Proceedings of the USENIX Annual Technical Conference (USENIX ATC '14)*, pp. 347–358, USENIX Association, Philadelphia, Pa, USA, June 2014.

[3] F. Lanze, A. Panchenko, I. Ponce-Alcaide, and T. Engel, "Undesired relatives: protection mechanisms against the evil twin attack in IEEE 802.11," in *Proceedings of the 10th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet '14)*, pp. 87–94, ACM, Québec, Canada, September 2014.

[4] J. Zhang, X. Zheng, Z. Tang et al., "Privacy leakage in mobile sensing: your unlock passwords can be leaked through wireless hotspot functionality," *Mobile Information Systems*, vol. 2016, Article ID 8793025, 14 pages, 2016.

[5] D. A. D. Zovi and S. A. Macaulay, "Attacking automatic wireless network selection," in *Proceedings of the 6th Annual IEEE System, Man and Cybernetics Information Assurance Workshop (SMC '05)*, pp. 365–372, West Point, NY, USA, June 2005.

[6] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[7] J. Herzen, R. Merz, and P. Thiran, "Distributed spectrum assignment for home WLANs," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 1573–1581, Turin, Italy, April 2013.

[8] O. Nakhila, E. Dondyk, M. F. Amjad, and C. Zou, "User-side Wi-Fi evil twin attack detection using SSL/TCP protocols," in *Proceedings of the 12th Annual IEEE Consumer Communications and Networking Conference (CCNC '15)*, pp. 239–244, IEEE, January 2015.

[9] S. Bratus, C. Cornelius, D. Kotz, and D. Peebles, "Active behavioral fingerprinting of wireless devices," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 56–61, New York, NY, USA, 2008.

[10] J. Cache, "Fingerprinting 802.11 implementations via statistical analysis of the duration field," *Uninformed.org*, vol. 5, 2006.

[11] D. McCoy, J. Franklin, J. Van Randwyk, D. Sicker, and P. Tabriz, Passive data-link layer 802.11 wireless device driver fingerprinting, January 2006.

[12] L. C. C. Desmond, C. C. Yuan, T. C. Pheng, and R. S. Lee, "Identifying unique devices through wireless fingerprinting," in *Proceedings of the ACM Conference on Wireless Network Security (WISEC '08)*, pp. 46–55, Alexandria, Va, USA, April 2008.

[13] C. Neumann, O. Heen, and S. Onno, "An empirical study of passive 802.11 device fingerprinting," in *Proceedings of the 32nd IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW '12)*, pp. 593–602, Macau, China, June 2012.

[14] R. Beyah, S. Kangude, G. Yu, B. Strickland, and J. Copeland, "Rogue access point detection using temporal traffic characteristics," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '04)*, vol. 4, pp. 2271–2275, Dallas, Tex, USA, November 2004.

[15] W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, and D. Towsley, "Passive online rogue access point detection using sequential hypothesis testing with TCP ACK-pairs," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC '07)*, pp. 365–378, ACM, San Diego, Calif, USA, October 2007.

[16] H. Han, B. Sheng, C. C. Tan, Q. Li, and S. Lu, "A timing-based scheme for rogue AP detection," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 11, pp. 1912–1925, 2011.

[17] C. D. Mano, A. Blaich, Q. Liao et al., "Ripps: rogue identifying packet payload slicer detecting unauthorized wireless hosts through network traffic conditioning," *ACM Transactions on Information and System Security*, vol. 11, no. 2, article no. 2, 2008.

[18] G. Qu and M. M. Nefcy, "RAPiD: an indirect rogue access points detection system," in *Proceedings of the IEEE 29th International Performance Computing and Communications Conference (IPCCC '10)*, pp. 9–16, IEEE, December 2010.

[19] K. S. A. P. Levis, "Rssi is under appreciated," in *Proceedings of the 3rd Workshop on Embedded Networked Sensors*, vol. 3031, p. 239242, Cambridge, Mass, USA, 2006.

[20] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," in *Proceedings of the 7th ACM Symposium on Modeling, Analysis*

and Simulation of Wireless and Mobile Systems (ACM MSWiM '04), pp. 78–82, ACM, Venice, Italy, October 2004.

[21] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2137–2148, 2003.

[22] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: decimeter level localization using wifi," *SIGCOMM Computer Communication Review*, vol. 45, no. 4, pp. 269–282, 2015.

[23] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, "FILA: fine-grained indoor localization," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 2210–2218, Orlando, Fla, USA, March 2012.

[24] 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, ACM, Istanbul, Turkey, August 2012.

[25] H. Liu, Y. Gan, J. Yang et al., "Push the limit of WiFi based localization for smartphones," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom '12)*, pp. 305–316, ACM, Istanbul, Turkey, August 2012.

[26] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding WiFi localization," in *Proceedings of the 11th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '13)*, pp. 249–262, ACM, Taipei, Taiwan, June 2013.

[27] J. Schulz-Zander, C. Mayer, B. Ciobotaru, S. Schmid, A. Feldmann, and R. Riggio, "Programming the home and enterprise WiFi with OpenSDWN," in *Proceedings of the ACM Conference on Special Interest Group on Data Communication (SIGCOMM '15)*, pp. 117–118, ACM, London, UK, August 2015.

[28] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: the insecurity of 802.11," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pp. 180–189, Rome, Italy, 2001.

[29] E. Tews, R. P. Weinmann, and A. Pyshkin, "Breaking 104 bit wep in less than 60 seconds," in *Proceedings of the Information Security Applications, International Workshop (Wisa '07)*, pp. 188–202, Jeju Island, Korea, August 2007.

[30] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, "Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[31] Q. Zhang, H. Zhong, L. T. Yang, Z. Chen, and F. Bu, "Privacy preserving highorder cfs algorithm on the cloud for clustering multimedia data," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 4s, pp. 66:1–66:15, 2016.

[32] 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.

[33] J. Wang, X. Chen, D. Fang, C. Q. Wu, Z. Yang, and T. Xing, "Transferring compressive-sensing-based device-free localization across target diversity," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2397–2409, 2015.

[34] J. Wang and D. Katabi, "Dude, where's my card? RFID positioning that works with multipath and non-line of sight," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '13)*, pp. 51–62, ACM, August 2013.

[35] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[36] S. Kullback, "Letter to the editor: the kullback-leibler distance," *The American Statistician*, vol. 41, no. 4, pp. 340–341, 1987.

[37] Y. Wen, X. Tian, X. Wang, and S. Lu, "Fundamental limits of RSS fingerprinting based indoor localization," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 2479–2487, Hong Kong, May 2015.

[38] H. Han, F. Xu, C. C. Tan, Y. Zhang, and Q. Li, "Defending against vehicular rogue aps," in *Proceedings of the IEEE INFOCOM*, pp. 1665–1673, April 2011.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 3906953, 12 pages
https://doi.org/10.1155/2017/3906953

Research Article

An Extended Technology Acceptance Model for Mobile Social Gaming Service Popularity Analysis

Hui Chen,[1,2] Wenge Rong,[1,2] Xiaoyang Ma,[3] Yue Qu,[2] and Zhang Xiong[1,2]

[1]*State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China*
[2]*School of Computer Science and Engineering, Beihang University, Beijing 100191, China*
[3]*Jacobs Institute, Cornell Tech, Cornell University, New York, NY, USA*

Correspondence should be addressed to Wenge Rong; w.rong@buaa.edu.cn

Received 23 September 2016; Accepted 1 December 2016; Published 3 January 2017

Academic Editor: Qingchen Zhang

The games industry has been growing prosperously with the development of information technology. Recently, with further advances in social networks and mobile services, playing mobile social gaming has gradually changed our daily life in terms of social connection and leisure time spending. What are the determinant factors which affect users intention to play such games? Therefore in this research we present an empirical study on WeChat, China's most popular mobile social network, and apply a technology acceptance model (TAM) to study the reasons beneath the popularity of games in mobile social networks. Furthermore, factors from social and mobile perspective are incorporated into the conventional TAM and their influence and relationships are studied. Experimental study on accumulated online survey data reveals several interesting findings and it is believed that this research offers the researchers in the community further insight in analysing the current popularity and future potential of mobile social games.

1. Introduction

With the development of information technology, video games have become one of the most important applications and are wildly popular with all kinds of people of all ages. Playing video games has gradually changed people's life style, particularly in terms of how leisure time is spent [1]. In addition, video games are also used to help people learn [2, 3], improve social skill [4], and even promote physical activity [5]. As an important video games platform, mobile devices, particularly smartphones, have become more and more popular. The increasing popularity of mobile devices is the key to open a huge market to mobile based gaming industry. For example, in China the mobile games players have increased to 279 million by early 2016 [6].

The smartphones provide a new platform for both social networking and video games. The mobile platform for social networks allows users to influence their friends [7] and have fun sharing their experiences [8]. Social networks have experienced exponential growth in recent years and with the further popularity of smartphones, mobile social network services will become one of the most popular applications due to their portability [9].

Due to the fact that playing games and using social networks are two of the most popular applications used daily on smartphones [10], it is worthwhile to investigate the integration of mobile social games. Currently playing games on mobile platform can have a lot of intentions, for example, education [11], while having fun in leisure time is a major purpose. In this research, we use games released on WeChat (Tencent's mobile social network service, known as Weixin (http://weixin.qq.com/) in China) as a case study to understand people's usage patterns and study what major determinants affect such games acceptance.

The WeChat App was first released in January 2011 as a mobile social network application which provides text, image, video, and voice messaging communication service. On 5th of August in 2013, Tencent released WeChat 5.0 which included a gaming centre. Several WeChat games were released with incredible numbers of games being downloaded. For example, a game called "Craz3 Match" was ranked 1st in App store just five hours after it was first released with more

than 20 million downloads over the following three days. Subsequently many more games have been released which were also top ranked after their initial release.

Given the popularity of WeChat based games, we study in depth the reasons behind their broad acceptance. Many techniques in the literature can be used to analyse such behaviour patterns and technology acceptance model (TAM) [12] is one of the leading approaches. During the past decades, researchers have successfully applied TAM and/or its extended models to explain user acceptance of many information technology based systems [13–15]. In the TAM model, several determinant factors, for example, usefulness and ease of use, have been identified as key influences of adoption of new information systems [16].

Besides these fundamental factors, there are other variables which also contribute to the popularity of WeChat games. A lot of previous researches on user's intention of using social networks and/or playing mobile games have been conducted and can provide inspiration in social network based game analysis. For example, an extended TAM model was proposed by D.-H. Shin and Y.-J. Shin to investigate the factors affecting user's acceptance of social network games [17]. Lin and Lu created a model to explain why people use social networks by integrating network externalities and motivation theory [18]. Another study by Liang and Yeh focuses on the effect of use contexts on the intention of continuing to play mobile games [19]. As for the mobile social gaming, Park et al. analysed some determinants of player acceptance and paid much attention to entertainment, mobility, connectedness, and sociability [20].

In this research, we try to explain why people continue to play mobile social games and investigate the main determinants and their relationships. Specifically, this work proposes an extended TAM model and adds several additional variables, such as social interaction, enjoyment, and altruism to enhance the understanding of user's intention to play such games. The evaluation and validation of the proposed model are conducted by analysing questionnaires accumulated online and several interesting findings are revealed.

The remainder of this paper is organised as follows. In Section 2 we will introduce the background of the TAM model and mobile social gaming. Section 3 will present the proposed extended TAM model and list the objectives and hypotheses. In Section 4 we will present the collection, processing, and analysis of the data and discuss the experimental results. Finally Section 5 concludes the paper and outlines possible future work.

2. Theoretical Foundations and Related Work

2.1. Technology Acceptance Model. In the area of information systems, there is a need for researchers to understand the reasons behind the users' actual usage of IT systems. To solve this problem, many technologies have been proposed, for example, Theory of Reasoned Action (TRA) [21], Model of Personal Computer Utilisation (MPCU) [22], Motivational Model (MM) [23], Unified Theory of Acceptance and Use of Technology (UTAUT) [24], Theory of Planned Behaviour (TPB) [25], and technology acceptance model (TAM) [12]. Of these approaches, technology acceptance model (TAM) has become one of the most popular and widely used techniques to elaborate on the rationality of users when they accept to use a certain information system. During the past decades, TAM has been successfully applied to lots of research domains and related applications and proven its capacity and validity in explaining user behaviour towards adoption of information systems.

In the earliest TAM model, it is argued that the actual system use is predicable by user motivation, which is also directly influenced by external variables, that is, system features, capabilities, and so on [12]. It is further suggested that user motivation consists of three influential factors, that is, perceived ease of use (PEOU), perceived usefulness (PU), and attitude towards using (ATT), which are able to explain the actual system use. In this TAM model, the attitude towards using, which is influenced by perceived usefulness (PU) and perceived ease of use (PEOU), is the major determinant for a user to accept or reject a certain system. Furthermore, perceived usefulness and perceived ease of use will be affected by several external stimuli. Davis finally hypothesised that perceived usefulness (PU) and perceived ease of use (PEOU) are the most important beliefs for a user to make a decision of whether to accept the system or not [12]. Since the TAM model was first proposed, it has been gradually refined and several other variables are added to the original TAM model, such as behavioural intention [16]. Because TAM has evolved into a leading model in predicting and explaining an information systems acceptance, it is believed the TAM model is also appropriate to analyse the popularity of mobile social gaming.

2.2. Mobile Social Gaming Analysis. Currently the video game has become one of the most important usages of advanced information technology. It has greatly transformed all people's behaviour pattern in spending their spare time [1], not only teenage but also elderly people [26]. Furthermore, with the development of the Internet, online multiplayer games are becoming more popular than single player games. As a result much effort has been devoted to understanding the popularity of online games. For example, Hsu and Lu tried to study the success of online games from the perspective of entertainment oriented technology and applied the TAM model by incorporating social influences and flow experience as belief-related constructs to predict the acceptance [27]. Lee argued that the flow experience is a more important factor than perceived enjoyment in influencing customers acceptance of online games [28] and further revealed that gender is a key moderator of online game acceptance. Later on Lee and Tsai proposed a theoretical research model, which integrates flow experience, human-computer interaction, social interaction, and perceived enjoyment, together with the technology acceptance model and Theory of Planned Behaviour to explain why people continue to play online games [29]. Wu and Liu suggested that trust is another important determinant for people continuing to play online games [30].

With the development of mobile devices, particularly the smartphone, playing online games in a mobile environment has become more and more popular as it extends the variance of place and time for users to play online games. According to the studies of Liu and Li, the effect of use context on the formation of users perceptions of mobile services is powerful [31]. Liang and Yeh used TAM to analyse mobile game acceptance and demonstrate that the use context has a significant moderating effect on people's intention to play mobile games [19]. Ha et al. conducted research on wireless mobile broadband games and argued that both technological and psychological aspects are of importance for mobile game adoption [32], by extending TAM to include flow experience and attractiveness and measure the moderating effects of gender and age. Similarly, Petrova and Qu studied the adoption of mobile gaming in New Zealand's youth market and their findings proposed that the expressiveness is the most significant influential factor affecting intention to play mobile games [33].

Social networks, such as Facebook, Twitter, and WeChat, have greatly changed our daily life [34]. Kwon et al. gave a comparative analysis of user acceptance of Facebook and Twitter by extended TAM model to find the key motivation factors in using social network services [35]. Rosen and Sherman extended TAM model with flow experience to explain the acceptance of people's intention to use social networks [36]. Lin and Lu found that the most influential factor affecting users in joining social network services is enjoyment, followed by number of peers and usefulness [18]. Their findings further suggest that gender difference also has different influences. Sledgianowski and Kulviwat also argued that playfulness and critical mass are strongest indicators of intent to use social networking websites [37]. Kwon and Wen applied the TAM model to construct an amended model which revealed three individual differences, that is, social identity, altruism, and telepresence [38]. Rauniar et al. added the factors of users critical mass, social networking site capability, and trustworthiness to extend the TAM model and the results provided evidence for the importance of additional key variables to TAM in considering user engagement on social media sites and other social media related business strategies [39]. Similarly, Kim et al. found that the major motives for using social network sites are seeking friends, social support, entertainment, information, and convenience [40].

From the discussion above, it is clear that playing online games and surfing social network services are the two major mobile applications. It is found that social network games have been widely implemented further into mobile devise as applications [41]. It is then becoming very interesting to ask what if these two applications are combined together? Social aspects are also important for gaming, not only in console gaming [42], but also in games on social networks [43]. For example, players can buy and sell virtual goods in games via social networks [44]. D.-H. Shin and Y.-J. Shin proposed an extended TAM model to investigate factors influencing user acceptance of social games [17]. They found that perceived playfulness and security have significant effects on game adoption. Their findings also revealed that flow experience plays a moderate role which affects various paths in the model. Lin et al. proposed a model to examine the determining factors of playing social games [45]. Their findings demonstrate that a state of arousal leads people to a higher level of continuing to play social games. Recently, considering the popularity of mobile social games, Park et al. investigated some factors which affect the intentions of users to play such games, for example, control, skill, mobility, and connectedness [20]. They found that satisfaction has a significant effect with multiple connections in the research model. Similarly, in a study by Wei and Lu, both network externalities and individual gratification significantly influence the intention to play social games on mobile devices [46]. They also proposed some factors such as time flexibility, but they appear to be less significant according to their investigation. Similarly, Ding et al. conducted an empirical study of mobile social games and found that mobility, desire for advancement, relationship building, escapism, and high engagement motivate players to enter the game and have recreational play [47].

3. Hypotheses

Mobile social gaming is a new platform for people to play games with other friends. In this research we will use WeChat games as a case study to understand such attraction. To understand the popularity of a game platform, a large number of factors can be attached for importance; for example, people may concern about privacy in social game as users could use personal information to buy equipment. Such information stored in the app's cloud environment is sensitive [48]. In this research, we mainly studied the factors from social perspective; as such we proposed an extended TAM model including traditional factors such as perceived ease of use (PEU), perceived usefulness (PU), attitude (ATT), and behaviour intention (BI). Moreover, a game is different from a regular IT system because its main purpose is for entertainment, enjoyment, and relaxation [32]. As a result in the proposed model, we added external variables, that is, perceived enjoyment (PE), use context (UC), and flow experience (FL), to provide understanding of pleasure and fun, which are often mentioned in previous studies [17, 31, 32]. In addition, since mobile social gaming is also a kind of social platform for users to share fun and other experience, we also added social interaction (SI) and altruism (ALT) into the proposed model [49, 50]. The research conceptual framework is depicted in Figure 1 and all variables and related hypotheses will be described in detail in the following subsections.

3.1. TAM. The proposed research model is an extension of the conventional TAM model. Therefore, the hypotheses of belief-attitude-intention-behaviour causal chain [21] is also adopted in the context of social based mobile games. Since games are entertainment oriented services, we use perceived ease of use (PEU) to represent how much effort a user thinks is needed to play a game. A high PEU score indicates that the game is easy to start playing and understand the

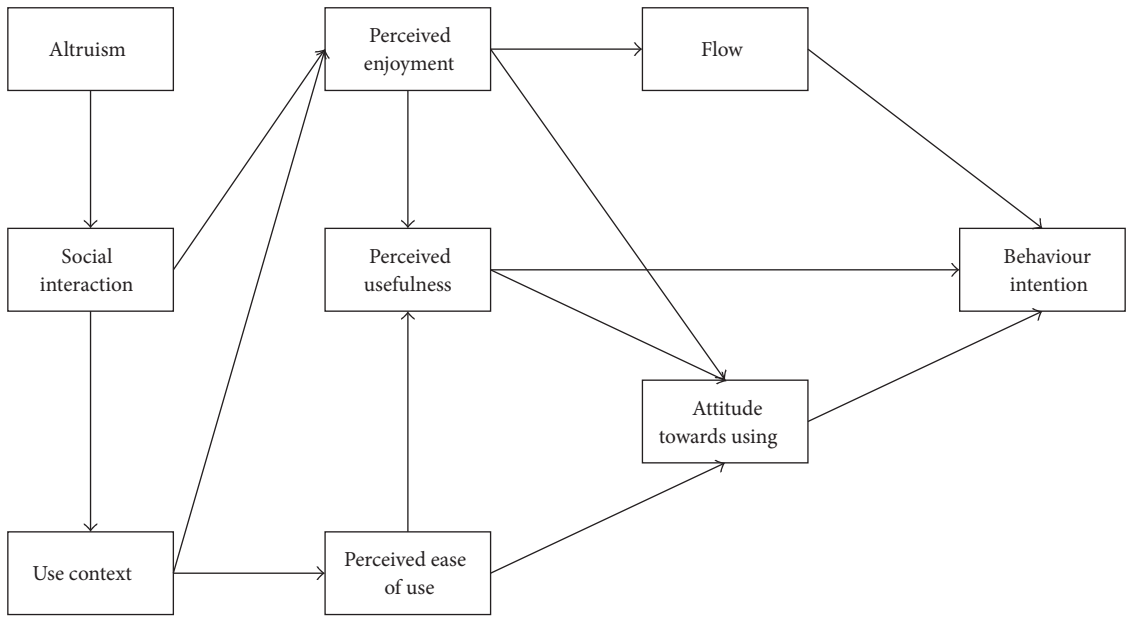


Figure 1: Proposed model.

rules. As a key structure in TAM, perceived usefulness (PU) has been refined and extended in various innovations [51], such as the improvement of job performance to measure innovation performance of "job/life/study" [31]. In this paper, we describe perceived usefulness as perceived improvement of the player's life experience caused by playing mobile social games. Consequently, we propose the following hypotheses:

(H1) Perceived ease of use (PEU) positively influences user's perceived usefulness (PU) of playing mobile social games.

(H2) Perceived ease of use (PEU) positively influences user's attitude (ATT) on social based mobile games.

(H3) Perceived usefulness (PU) positively influences user's attitude (ATT) on mobile social gaming.

(H4) Perceived usefulness (PU) positively influences user's intention (BI) to play mobile social games.

(H5) Attitude (ATT) positively influences user's intention (BI) to play mobile social games.

3.2. Perceived Enjoyment. Perceived enjoyment (PE) is the extent to which an activity is perceived to be enjoyable without considering any performance consequences [23]. It is an intrinsic motivation referring to the pleasure and satisfaction from performing a behaviour [52]. van der Heijden indicated that perceived enjoyment has a significant and positive influence on people's attitude and intention towards website adoption [53]. Moreover, in a study of the determinants of adoption of mobile games under mobile broadband wireless access environment, Ha et al. argued that perceived enjoyment should be one of the characteristics of games and perceived enjoyment should be included when analysing game systems [32]. Since social network based mobile gaming is also a kind of hedonic systems, we made the following hypotheses:

(H6) Perceived enjoyment (PE) positively influences attitude (ATT) on mobile social gaming.

(H7) Perceived enjoyment (PE) positively influences intention (FL) to play mobile social games.

(H8) Perceived enjoyment (PE) positively perceived usefulness (PU) of playing mobile social gaming.

3.3. Use Context. Use context (UC) refers to the environment where the technology will be used [54]. Use context is not just a point in time and space in which a particular action is taken. It also relates to situational and social contexts. Since smartphones have become a daily necessity in people's life, users may have a positive attitude towards a service when it fits a certain use context. Therefore, contextual factors should be added to traditional TAM model when studying user acceptance of mobile services. In fact, many previous studies have tried to integrate use context to extend the model. For example, in a study of exploring consumer adoption of mobile payments, Mallat pointed out that the adoption of mobile payment relies on certain situational factors such as a lack of other payment methods [55].

As many contextual factors may have great effect on user adoption, our study focuses on two factors which are highly related to mobile games, that is, the place where the people are and how the people feel at that time. For example, when people are in crowded public transportation and they feel bored, using a laptop is not possible but there is space to use a mobile phone. People can play mobile games to pass the time

and enjoy themselves. Hence, two hypotheses are posited as follows:

(H9) Use context (UC) positively influences perceived enjoyment (PE) of mobile social gaming.

(H10) Use context (UC) positively influences perceived ease of use (PEU) to play mobile social games.

3.4. Flow. Flow (FL) was first put forward by M. Csikszentmihalyi and I. Csikszentmihalyi and defined as the holistic experience when involved in the action [56]. Due to the complexity and multidimensionality of flow [57], it has been extensively applied in a wide range of contexts, such as sports, shopping, rock climbing, dancing, and gaming [58]. Ghani argued that flow can be measured by enjoyment and concentration and found that perceived control and challenges can predict flow [59]. In subsequent studies, Li and Browne further explained flow with four dimensions: focused attention, control, curiosity, and temporal dissociation [60]. In this study, the concept of flow focuses on curiosity. Curiosity refers to the situation that people stay curious about the system and try to accomplish technological competence while being engaged in an action [14]. In Moon and Kim's study, users remain curious about the Internet because they can acquire new information and knowledge. As for mobile social gaming, people can not only play games but also compete and share with their friends. These above remain the curiosity of the players and lead to the replay intention. Consequently, we have the following hypothesis:

(H11) Flow (FL) positively influences intention (BI) to play mobile social gaming.

3.5. Social Interaction. Interaction is a kind of behaviour between two or more objects. In prior studies interaction is usually classified into two types. The first is the interaction between the user and the system, and the second is user-to-user interaction [61]. In this study, since we focus on mobile social networks, we focus on user-to-user interaction which is usually called social interaction (SI). Social games are built to be enjoyed and shared with friends through existing social networks and platforms. In WeChat, people play games on the same platform which allows them to share scores and compete with each other. Furthermore, people can give lives as present to friends, which causes closer relationship. Those above really bring much fun and therefore we propose the following hypotheses:

(H12) Social interaction (SI) positively influences perceived enjoyment (PE) of mobile social gaming.

(H13) Social interaction (SI) positively influences use context (UC) of mobile social gaming.

3.6. Altruism. Altruism (AL) can be classified into kin altruism and reciprocal altruism [38]. Kin altruism refers to concept that people sacrifice their own benefits to help their genetic relatives, and reciprocal altruism means that people help others because they believe that they will receive similar assistance in return some day in the future [62]. It is interesting that users display both kinds of altruism in mobile social gaming. In the popular WeChat game "Aircraft Wars," people can give their own lives to friends as a present and this is also popular in other WeChat games. However, altruism is an alternative to explain the people's behaviour [63]. It is difficult to understand the altruistic behaviour from the traditional economic view that people behave to maximise their own preferences [45]. Considering that TAM is an extension of the Theory of Reasoned Action (TRA), it is not suitable to apply altruism to traditional TAM framework [64]. We should add a new perceived construct to explain the altruistic behaviour. In the context of mobile social gaming, the altruistic behaviour may be more motivated by the perceived enjoyment of the players due to the friendship between them. Hence, our hypothesis on altruism is stated as follows:

(H14) Altruism (ALT) positively influences social interaction (SI) of mobile social gaming.

Table 1: Design of the questionnaire.

Factor	Abbreviation	Question number	Verification questions (Y/N)
Social interaction	SI	3	N
Altruism	ALT	3	N
Perceived enjoyment	PE	4	Y
Perceived usefulness	PU	3	N
Perceived ease of use	PEU	3	Y
Flow	FL	3	N
Attitude	ATT	3	N
Use context	UC	3	N
Behaviour intention	BI	3	N

4. Results and Analysis

4.1. Data Collection. In this research we published questionnaires on an online survey agency to collect the experimental data. The original questionnaire consists of two parts. The first part has 8 questions to collect the basic information of the informants, such as sex, age, and use experience with WeChat and/or games. The second part is the main component of the questionnaire and consists of 32 questions to investigate the 9 factors introduced in previous section. Each question is measured on a 7-point Likert scale with the end points of "strongly agree (7)" and "strongly disagree (1)".

The data collection process uses a two-step approach. Firstly we conducted a pilot test to verify the questionnaire's accuracy, which results in the removal of 4 questions from the original questionnaire. As a result the final questionnaire consists of 28 questions, among which two questions are designed as reverse questions to help judge insincere responses. Table 1 lists the final published questionnaire, and the 26 questions (without the two reverse questions) are listed in Table 2.

Table 2: Questionnaire.

Factor	Item	Measure
Social interaction (SI)	SI1	I like to play the game which my friends play.
	SI2	WeChat games provide a platform for me to play games with my friends.
	SI3	I like to play games with friends.
Altruism (ALT)	ALT1	I will give my friends gifts or other in-game help.
	ALT2	I often help my friends when they need help in WeChat games.
	ALT3	My friends often give me feedback when I offer help they need in WeChat games.
Perceived enjoyment (PE)	PE1	It is interesting to play WeChat games.
	PE2	Playing WeChat games brings enjoyment to my daily life.
	PE3	I always feel happy when I am playing WeChat games.
Perceived usefulness (PU)	PU1	Playing WeChat games makes my life different.
	PU2	Playing WeChat games makes my life better.
	PU3	Playing WeChat games is useful for me.
Perceived ease of use (PEU)	PEU1	It is easy for me to play WeChat games.
	PEU2	It is easy for me to master the rules of the games.
Flow (FL)	FL1	I will not be tired of WeChat games in a short time.
	FL2	I will not lose interest in WeChat games in a short time.
	FL3	It happened often for me to ignore the time past when I play WeChat games.
Attitude (ATT)	ATT1	It is a good idea for me to play WeChat games during my free time.
	ATT2	I feel good towards WeChat games.
	ATT3	I like playing WeChat games.
Use context (UC)	UC1	Playing WeChat games is a way to spend free time for me.
	UC2	I will consider to play WeChat games when I am bored.
	UC3	I will consider to play WeChat games when I have free time.
Behaviour intention (BI)	BI1	I want to play more kinds of WeChat games later.
	BI2	I will keep playing WeChat games.
	BI3	I will play WeChat games with my friends together.

Table 3: Data filtering result.

Item	Number
Total responses	491
Not played WeChat games	122
Insincere response	61
Effective responses	308

A total of 491 responses were collected from the online survey. In order to improve the quality of the data we filter out responses which fit the following criteria: (1) Eliminate the responses of respondents who have never played a WeChat game. (2) Eliminate the insincere responses through data filtering on the two verification questions. (3) Eliminate the insincere responses which look like "Straight-Line" or "Wave" [65]. The final result is as shown in Table 3. In the field of human-computer interaction for qualitative analysis, the size of the data set containing more than 200 valid responses can be viewed as an effective data set [51]. In this experiment, we collected 308 valid questionnaire responses so we regard this as an effective data set.

4.2. Data Analysis

4.2.1. Reliability Analysis. In order to analyse the effectiveness of the original data, the first step of the experiment is to conduct data standardisation. In this step we calculate the average and standard deviation of each question result and also the average for each category. The results are shown in Table 4. From the table we can see that the average of all factors is greater than 5, which suggests that the assumptive factors were typical.

Afterwards we further employ Cronbach's alpha coefficient to show the convergent validity and internal reliability of the factors, which are listed in Table 5. From Table 5 we can see that the total Cronbach's alpha coefficient is 0.947 and the coefficients of each factor are greater than 0.7. It is then argued that the total Cronbach's alpha coefficient is acceptable (>0.8 [66]), and the coefficients of each factor are also acceptable (>0.7 [66]). As a result we conclude that the data are reliable measures for their factors.

Meanwhile, discriminant validity is verified as to ensure that variables relate more strongly to their own factor than to other factors. As shown in Table 6, the maximum correlations between different factors are below 0.70 [32]. Therefore it is

Table 4: Question standardisation and reliability analysis.

Factor	Question	AVG	SD	AVG
SI	SI1	5.98	0.943	5.88
	SI2	5.86	1.040	
	SI3	5.81	1.085	
ALT	ALT1	5.94	1.003	5.92
	ALT2	5.97	0.920	
	ALT3	5.87	0.947	
PE	PE1	5.87	0.989	5.84
	PE2	5.86	0.909	
	PE3	5.80	0.958	
PU	PU1	5.36	1.220	5.37
	PU2	5.41	1.153	
	PU3	5.36	1.305	
PEU	PEU1	5.93	0.773	5.98
	PEU2	6.03	0.786	
FL	FL1	5.32	1.265	5.44
	FL2	5.50	1.035	
	FL3	5.51	1.163	
ATT	ATT1	5.87	0.946	5.84
	ATT2	5.85	0.939	
	ATT3	5.79	0.929	
UC	UC1	6.04	0.855	5.99
	UC2	6.01	0.941	
	UC3	5.92	0.963	
BI	BI1	5.98	0.950	5.94
	BI2	6.05	0.910	
	BI3	5.80	1.080	

able to conclude that the factors are sufficiently distinct and uncorrelated.

4.2.2. Principal Component Analysis. The next step of data analysis is to conduct principal component analysis (PCA). Before that, it is necessary to test the adequacy of data. In this research, KMO Testing and Bartlett Testing are employed to validate whether the data are suitable for PCA process [67]. The result is shown in Table 7. As suggested by commonly used KMO measures, it is concluded that our collected data are appropriate for principal component analysis. After the PCA process, the next step is to rotate the matrix from PCA analysis to distinguish the importance of the 9 factors. The result is shown in Table 8 and it is seen that the importance rank of the 9 factors, from high to low, is PU, ATT, SI, ALT, PE, UC, FL, BI, and PEU, respectively.

4.3. Hypothesis Evaluation

4.3.1. Model Fit Indices. To evaluate the proposed model and validate the proposed hypotheses, eight fit indices are employed in this research, that is, X^2, GFI, AGFI, RMSEA, RMR, CFI, NFI, and IFI [68]. The fitness results for the measurement are shown in Table 9 and each of the fitness measures is acceptable. Consequently, all the measures chosen in this work appear to show that the proposed model can provide a good fit to the data, thereby making it possible to conduct path analysis for the proposed model.

Table 5: Cronbach's alpha coefficient of each factor.

Factor	Cronbach's alpha coefficient
SI	0.789
ALT	0.776
PE	0.799
PU	0.872
PEU	0.718
FL	0.749
ATT	0.786
UC	0.727
BI	0.735
Total (26 questions)	0.947

4.3.2. Path Analysis. The aim of path analysis is to evaluate the veracity and reliability of the hypothetical model and measure the strength of the causal relationship between variables. We examined the structural equation model by testing the hypothesised relationships between various factors, as shown in Figure 2 and Table 10.

4.4. Discussion. This study developed a theoretical framework and discussed the structural equation modelling analysis of the proposed theoretical framework for mobile social game adoption. Consistent with previous studies focusing on online games and mobile social network services [17, 20, 45–47], our findings in this study provided empirical support for the proposed TAM extended model. The results clarified our understanding of people's attitudes and intentions towards playing mobile social games and also helped to reveal implications for the successful implementation of WeChat games in China. The measurement of this study provided a good fit to the data, thereby lending support to the proposed model. Overall, the results show that the proposed model is able to accurately describe the intentions of users to play mobile social games.

From this study it is found that perceived enjoyment and perceived ease of use are the chief determinants of user attitudes to play mobile social games. This may suggest that (1) players regard the level of enjoyment from playing mobile social games as the most significant factor and (2) players prefer to play some easier to get started mobile social games which would not cost them much effort. Of these two factors, perceived enjoyment shows a much stronger effect than perceived ease of use, which implies that entertainment oriented technologies will be paid much attention by the markets. Furthermore, this model shows insignificant role of perceived usefulness, which sharply contrasts perceived enjoyment and perceived ease of use, in affecting user attitude to play mobile social games. From this research it is concluded

Table 6: Intercorrelations between factors.

	SI	ALT	PE	PU	PEU	FL	ATT	UC	BI
SI	1.000								
ALT	0.612	1.000							
PE	0.637	0.492	1.000						
PU	0.527	0.470	0.417	1.000					
PEU	0.530	0.479	0.562	0.292	1.000				
FL	0.651	0.509	0.631	0.621	0.494	1.000			
ATT	0.563	0.452	0.475	0.602	0.379	0.598	1.000		
UC	0.657	0.508	0.618	0.568	0.518	0.687	0.638	1.000	
BI	0.695	0.556	0.674	0.603	0.541	0.673	0.600	0.699	1.000

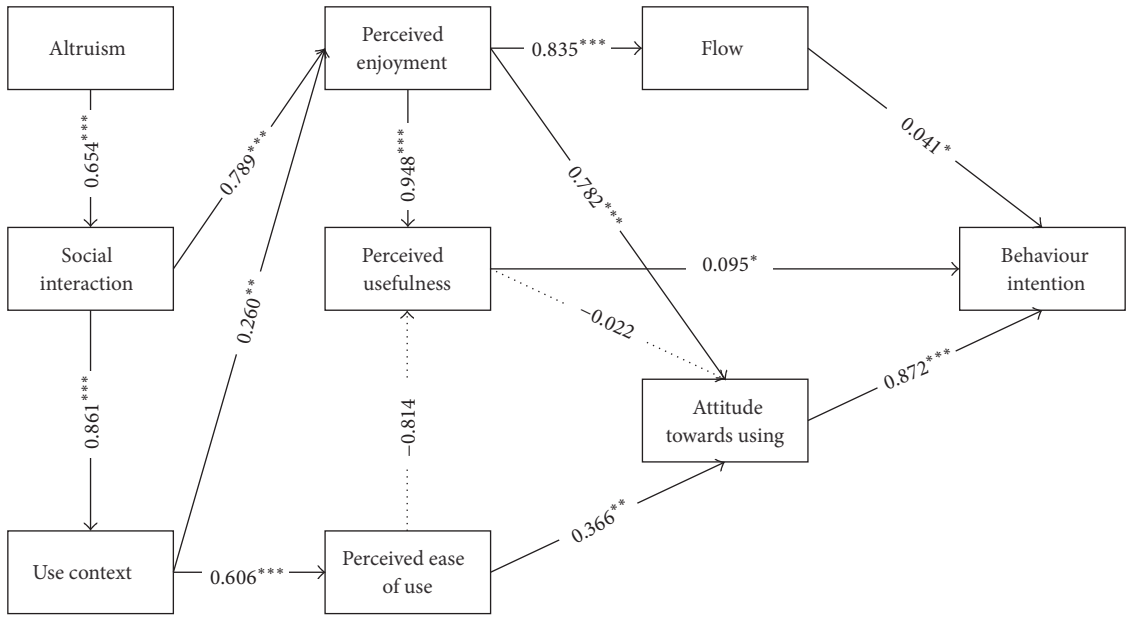


Figure 2: Path verification. $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Table 7: KMO and Bartlett Testing.

Kaiser-Meyer-Olkin	.942
Bartlett Testing	
X^2	4518.333
df	325
Sig.	.000

that perceived usefulness also does not have very strong effect on the actual behaviour intention, which corroborates previous studies [17, 27] that perceived usefulness may have an insignificant effect on user attitude. Therefore combining our findings with other existing works in the literature, it may be inferred that, in the domain of mobile social games, users mainly want to easily get fun from social games as a hedonic system in mobile environment anytime and anywhere without considering too many performance consequences.

Considering the importance and the significance of perceived enjoyment, it is deserved to conduct further investigation to study the relationship between it with other factors. From this research, it is reasonable to argue that enjoyment can enhance perception of flow. In fact, the popularity of some WeChat games is partly because of its mechanism of making fun from keeping playing to beat friends. However, due to the fact that normally users play WeChat games to kill the boring time, for example, when using public transportation, it is not surprising to see that flow does not exert significant effect on the intention.

Our findings also shows that social interaction does have strong influence on perceived enjoyment while it also has significant influence on use context. Mobile social games provide a new platform for users to communicate with each other and then close the relationship among them. For example, in WeChat games, users can compete against, offer help to, and/or interact promptly with their friends, thereby making the gaming more interesting. In this research, it is also found that social interaction in WeChat games is also supported by altruism. Offering help in the games does bring

Table 8: Rotation matrix.

	Component								
	1	2	3	4	5	6	7	8	9
SI1	0.106	0.339	**0.578**	0.246	0.129	0.322	0.086	0.037	0.149
SI2	0.227	0.151	**0.713**	0.212	0.196	0.040	0.195	0.189	0.074
SI3	0.190	0.167	**0.766**	0.129	0.199	0.152	0.080	0.083	0.070
ALT1	0.227	0.104	0.189	**0.689**	−0.019	0.125	0.213	0.340	0.114
ALT2	0.043	0.165	0.114	**0.755**	0.198	0.310	0.071	−0.066	0.023
ALT3	0.211	0.106	0.171	**0.770**	0.180	−0.026	−0.034	0.105	0.183
PE1	0.279	0.322	0.320	0.096	**0.483**	0.108	0.207	0.400	−0.005
PE2	0.255	0.188	0.237	0.145	**0.707**	0.200	0.191	0.023	0.120
PE3	0.232	0.175	0.219	0.230	**0.598**	0.092	0.086	0.174	0.266
PU1	**0.767**	0.183	0.222	0.103	0.096	−0.005	0.243	0.302	−0.009
PU2	**0.778**	0.226	0.118	0.115	0.309	0.134	0.130	−0.009	−0.010
PU3	**0.826**	0.118	0.168	0.211	0.136	0.128	0.102	0.031	0.146
PEU1	0.108	0.037	0.079	0.090	0.222	0.201	0.128	0.048	**0.840**
PEU2	−0.034	0.401	0.160	0.296	0.050	0.157	−0.011	0.225	**0.613**
FL1	0.176	0.117	0.135	0.072	0.127	0.019	**0.843**	0.260	0.021
FL2	0.253	0.281	0.133	0.089	0.196	0.221	**0.730**	−0.154	0.163
FL3	0.383	0.262	0.192	0.307	−0.077	0.132	**0.571**	0.038	0.116
ATT1	0.130	**0.505**	0.121	0.163	0.398	0.163	0.177	0.485	0.125
ATT2	0.209	**0.669**	0.216	0.024	0.288	0.016	0.284	0.067	0.075
ATT3	0.199	**0.589**	0.195	0.239	0.315	0.209	0.004	0.156	0.134
UC1	0.080	0.362	0.176	0.133	0.454	**0.504**	0.122	0.021	0.228
UC2	0.074	0.117	0.013	0.206	0.133	**0.751**	0.151	0.233	0.151
UC3	0.148	0.116	0.387	0.053	0.152	**0.656**	−0.001	0.101	0.140
BI1	0.235	0.366	0.238	0.136	0.308	0.312	0.220	**0.410**	0.112
BI2	0.107	0.064	0.175	0.181	0.065	0.457	0.046	**0.664**	0.195
BI3	0.349	0.061	0.343	0.118	0.312	0.243	0.140	**0.488**	0.093

Table 9: Fit indices for the measurement.

	Results	Recommended criteria
X^2	1.844	<5.0
GFI	0.886	>0.85, close to 1
AGFI	0.854	>0.80, close to 1
RMSEA	0.052	≤0.06, close to 0
RMR	0.043	≤0.08, close to 0
CFI	0.947	>0.90, close to 1
NFI	0.892	>0.85, close to 1
IFI	0.947	>0.90, close to 1

Table 10: Analysis of significance of path coefficient.

Hypothesis	Estimate	Supported?
(H1) PEU→PU	−0.814	N
(H2) PEU→ATT	0.366**	Y
(H3) PU→ATT	−0.022	N
(H4) PU→BI	0.095*	Y
(H5) ATT→BI	0.872***	Y
(H6) PE→ATT	0.782***	Y
(H7) PE→FL	0.835***	Y
(H8) PE→PU	0.948***	Y
(H9) UC→PE	0.260**	Y
(H10) UC→PEU	0.606***	Y
(H11) FL→BI	0.041*	Y
(H12) SI→PE	0.789***	Y
(H13) SI→UC	0.861***	Y
(H14) ALT→SI	0.654***	Y

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.

a lot of fun and social reputation among friends. As such it suggests that social interaction plays a key role in increasing the enjoyment, thereby increasing the user attitude to play WeChat games.

Meanwhile, since users can use WeChat to communicate with each other when they have spare time, it is easier for a user to realise other friends' activity in WeChat games with portable smartphones. The use context for easily accessing and playing mobile social games by social connection does provide more chance for users to get involved into WeChat games, which is also the major cause of WeChat games spreading. This result supports previous research on use context [19, 31, 69]. Due to the fact that mobile social games have low requirement on network environment, hard devices,

and storage space, it is easy for people to play anytime and anywhere.

This proposed extended TAM model has several practical and theoretical implications for researchers and engineers to develop popular mobile social games. This study provided some in-depth analysis of popularity of WeChat games in China and then can be applied into development of games industry. It is argued that successful mobile social games should exert significant efforts to deliver enjoyable games in an easily accessible way as well as to provide excellent social interaction experience to encourage users to share their fun.

5. Conclusion and Future Work

Nowadays along with the development of social network service and mobile devices, social network based mobile gaming has become wildly popular. In this research we provide a use case analysis of the factors affecting acceptance of mobile social games on WeChat. To this end, we employ a technology acceptance model and integrate some amending predictors from social and mobile perspective. Our analysis of over 300 valid questionnaire respondents provides revealing findings on the influence of 9 factors on the acceptance of mobile social games. We believe that this research provides invaluable insight for mobile social game service providers, enabling better understanding of adoption behaviour and thus further improving their services.

Similar to other researches, there are several limitations in this study which deserve future effort to address. The major issue is related to the users of WeChat. The questionnaire in this research is in Chinese and all responses are from Mainland China. Furthermore, WeChat is not the only service for the social network though it is the most popular one in China indeed. Using WeChat as case study in this paper does provide some interesting findings; however, the results may be not easy to generalise. It would be interesting to extend this work into an international context and perhaps consider other social networks.

In contrast with other studies, there may be some important factors which may significantly contribute to the integrated model and deserve to be further investigated. For example, considering the possibility for WeChat games to involve payment and advertisement in terms of virtual gift, it can be forecast that user's comprehensive sense of security would have significant influence on user attitudes towards to mobile social games, thereby making perceived security an essential factor for further study. Furthermore, continuous usage of mobile games is also important as attracting users to use a game is a challenge but keeping the users to play with games is another even more challenging task. Therefore, analysis of factors for mobile game's continuous usage deserves to be studied further in the future research.

Competing Interests

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

Acknowledgments

This work was supported by the State Key Laboratory of Software Development Environment of China (no. SKLSDE-2015ZX-23), the National Natural Science Foundation of China (no. 61472021), and the Fundamental Research Funds for the Central Universities.

References

[1] T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," *Computers & Education*, vol. 59, no. 2, pp. 661–686, 2012.

[2] M. Papastergiou, "Digital game-based learning in high school computer science education: impact on educational effectiveness and student motivation," *Computers & Education*, vol. 52, no. 1, pp. 1–12, 2009.

[3] G.-J. Hwang and P.-H. Wu, "Advancements and trends in digital game-based learning research: a review of publications in selected journals from 2001 to 2010," *British Journal of Educational Technology*, vol. 43, no. 1, pp. E6–E10, 2012.

[4] A. M. Piper, E. O'Brien, M. R. Morris, and T. Winograd, "SIDES: a cooperative tabletop computer game for social skills development," in *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pp. 1–10, 2006.

[5] A. Barnett, E. Cerin, and T. Baranowski, "Active video games for youth: a systematic review," *Journal of Physical Activity and Health*, vol. 8, no. 5, pp. 724–737, 2011.

[6] CNNIC, "35th statistical report on Internet development in China," 2016, https://cnnic.com.cn/IDR/ReportDownloads/201604/P020160419390562421055.pdf.

[7] C. López-Nicolás, F. J. Molina-Castillo, and H. Bouwman, "An assessment of advanced mobile services acceptance: contributions from TAM and diffusion theory models," *Information & Management*, vol. 45, no. 6, pp. 359–364, 2008.

[8] D. M. Boyd and N. B. Ellison, "Social network sites: definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, pp. 210–230, 2007.

[9] S. Nikou and H. Bouwman, "Ubiquitous use of mobile social network services," *Telematics and Informatics*, vol. 31, no. 3, pp. 422–433, 2014.

[10] PwC, "Mobile advertising in China: what do Chinese consumers want and how should businesses be engaging with them?" 2014, http://www.pwccn.com/webmedia/doc/635358539404587393_mobile_ad_china_cut_may2014.pdf.

[11] G. Schwabe and C. Göth, "Mobile learning with a mobile game: design and motivational effects," *Journal of Computer Assisted Learning*, vol. 21, no. 3, pp. 204–216, 2005.

[12] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quarterly*, vol. 13, no. 3, pp. 319–340, 1989.

[13] J. C.-C. Lin and H. Lu, "Towards an understanding of the behavioural intention to use a web site," *International Journal of Information Management*, vol. 20, no. 3, pp. 197–208, 2000.

[14] J.-W. Moon and Y.-G. Kim, "Extending the TAM for a world-wide-web context," *Information & Management*, vol. 38, no. 4, pp. 217–230, 2001.

[15] M. Koufaris, "Applying the technology acceptance model and flow theory to online consumer behavior," *Information Systems Research*, vol. 13, no. 2, pp. 205–223, 2002.

[16] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, "User acceptance of computer technology: a comparison of two theoretical models," *Management Science*, vol. 35, no. 8, pp. 982–1003, 1989.

[17] D.-H. Shin and Y.-J. Shin, "Why do people play social network games?" *Computers in Human Behavior*, vol. 27, no. 2, pp. 852–861, 2011.

[18] K.-Y. Lin and H.-P. Lu, "Why people use social networking sites: an empirical study integrating network externalities and motivation theory," *Computers in Human Behavior*, vol. 27, no. 3, pp. 1152–1161, 2011.

[19] T.-P. Liang and Y.-H. Yeh, "Effect of use contexts on the continuous use of mobile services: the case of mobile games," *Personal and Ubiquitous Computing*, vol. 15, no. 2, pp. 187–196, 2011.

[20] E. Park, S. Baek, J. Ohm, and H. J. Chang, "Determinants of player acceptance of mobile social network games: an application of extended technology acceptance model," *Telematics and Informatics*, vol. 31, no. 1, pp. 3–15, 2014.

[21] M. Fishbein and I. Ajzen, *Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research*, Addison-Wesley, 1975.

[22] R. L. Thompson, C. A. Higgins, and J. M. Howell, "Personal computing: toward a conceptual model of utilization," *MIS Quarterly*, vol. 15, no. 1, pp. 125–143, 1991.

[23] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, "Extrinsic and intrinsic motivation to use computers in the workplace," *Journal of Applied Social Psychology*, vol. 22, no. 14, pp. 1111–1132, 1992.

[24] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User acceptance of information technology: toward a unified view," *MIS Quarterly*, vol. 27, no. 3, pp. 425–478, 2003.

[25] I. Ajzen, "The theory of planned behavior," *Organizational Behavior and Human Decision Processes*, vol. 50, no. 2, pp. 179–211, 1991.

[26] Q. Wang and X. Sun, "Investigating gameplay intention of the elderly using an extended technology acceptance model (ETAM)," *Technological Forecasting and Social Change*, vol. 107, pp. 59–68, 2016.

[27] C.-L. Hsu and H.-P. Lu, "Why do people play on-line games? An extended TAM with social influences and flow experience," *Information & Management*, vol. 41, no. 7, pp. 853–868, 2004.

[28] M.-C. Lee, "Understanding the behavioural intention to play online games: an extension of the theory of planned behaviour," *Online Information Review*, vol. 33, no. 5, pp. 849–872, 2009.

[29] M. Lee and T. Tsai, "What drives people to continue to play online games? An extension of technology model and theory of planned behavior," *International Journal of Human-Computer Interaction*, vol. 26, no. 6, pp. 601–620, 2010.

[30] J. Wu and D. Liu, "The effects of trust and enjoyment on intention to play online games," *Journal of Electronic Commerce Research*, vol. 8, no. 2, pp. 128–140, 2007.

[31] Y. Liu and H. Li, "Exploring the impact of use context on mobile hedonic services adoption: an empirical study on mobile gaming in China," *Computers in Human Behavior*, vol. 27, no. 2, pp. 890–898, 2011.

[32] I. Ha, Y. Yoon, and M. Choi, "Determinants of adoption of mobile games under mobile broadband wireless access environment," *Information & Management*, vol. 44, no. 3, pp. 276–286, 2007.

[33] K. Petrova and H. Qu, "Playing mobile games: consumer perceptions: an empirical study," in *Proceedings of the International Conference on e-Business*, pp. 209–214, 2007.

[34] K.-Y. Lin and H.-P. Lu, "Predicting mobile social network acceptance based on mobile value and social influence," *Internet Research*, vol. 25, no. 1, pp. 107–130, 2015.

[35] S. J. Kwon, E. Park, and K. J. Kim, "What drives successful social networking services? A comparative analysis of user acceptance of Facebook and Twitter," *The Social Science Journal*, vol. 51, no. 4, pp. 534–544, 2014.

[36] P. Rosen and P. Sherman, "Hedonic information systems: acceptance of social networking websites," in *Proceedings of the 12th Americas Conference on Information Systems*, p. 162, Acapulco, Mexico, August 2006.

[37] D. Sledgianowski and S. Kulviwat, "Social network sites: antecedents of user adoption and usage," in *Proceedings of the 14th Americas Conference on Information Systems*, p. 83, Toronto, Canada, August 2008.

[38] O. Kwon and Y. Wen, "An empirical study of the factors affecting social network service use," *Computers in Human Behavior*, vol. 26, no. 2, pp. 254–263, 2010.

[39] R. Rauniar, G. Rawski, J. Yang, and B. Johnson, "Technology acceptance model (TAM) and social media usage: an empirical study on Facebook," *Journal of Enterprise Information Management*, vol. 27, no. 1, pp. 6–30, 2014.

[40] Y. Kim, D. Sohn, and S. M. Choi, "Cultural difference in motivations for using social network sites: a comparative study of American and Korean college students," *Computers in Human Behavior*, vol. 27, no. 1, pp. 365–372, 2011.

[41] C. Feijoo, J.-L. Gómez-Barroso, J.-M. Aguado, and S. Ramos, "Mobile gaming: industry challenges and policy implications," *Telecommunications Policy*, vol. 36, no. 3, pp. 212–221, 2012.

[42] A. Voida and S. Greenberg, "Wii all play: the console game as a computational meeting place," in *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, pp. 1559–1568, Boston, Mass, USA, April 2009.

[43] J. Kim, Y. Chang, and M. Park, "Why do people like to play social network games with their friends? A focus on sociability and playability," in *Proceedings of the 17th Pacific Asia Conference on Information Systems*, p. 78, 2013.

[44] J. Hamari and L. Keronen, "Why do people buy virtual goods? A literature review," in *Proceedings of the 49th Hawaii International Conference on System Sciences*, pp. 1358–1367, Koloa, Hawaii, USA, January 2016.

[45] T. Lin, H. Lu, H. Hsu, S. Hsing, and T. Ho, "Why do people continue to play social network game (SNG)? An empirical study by social and emotional perspectives," *International Journal of E-Adoption*, vol. 5, no. 4, pp. 22–35, 2013.

[46] P.-S. Wei and H.-P. Lu, "Why do people play mobile social games? An examination of network externalities and of uses and gratifications," *Internet Research*, vol. 24, no. 3, pp. 313–331, 2014.

[47] Y. Ding, Y. Zhou, and A. Kankanhalli, "Why do I invite friends to join: an empirical study of mobile social network game," in *Proceedings of the 18th Pacific Asia Conference on Information Systems*, p. 137, June 2014.

[48] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[49] S. Trepte, L. Reinecke, and K. Juechems, "The social side of gaming: how playing online computer games creates online and offline social support," *Computers in Human Behavior*, vol. 28, no. 3, pp. 832–839, 2012.

[50] O. Curry, S. G. B. Roberts, and R. I. M. Dunbar, "Altruism in social networks: evidence for a 'kinship premium,'" *British Journal of Psychology*, vol. 104, no. 2, pp. 283–295, 2013.

[51] P. Legris, J. Ingham, and P. Collerette, "Why do people use information technology? A critical review of the technology acceptance model," *Information & Management*, vol. 40, no. 3, pp. 191–204, 2003.

[52] J. Doll and I. Ajzen, "Accessibility and stability of predictors in the theory of planned behavior," *Journal of Personality and Social Psychology*, vol. 63, no. 5, pp. 754–765, 1992.

[53] H. van der Heijden, "Factors influencing the usage of websites: the case of a generic portal in The Netherlands," *Information & Management*, vol. 40, no. 6, pp. 541–549, 2003.

[54] L. van de Wijngaert and H. Bouwman, "Would you share? Predicting the potential use of a new technology," *Telematics and Informatics*, vol. 26, no. 1, pp. 85–102, 2009.

[55] N. Mallat, "Exploring consumer adoption of mobile payments—a qualitative study," *The Journal of Strategic Information Systems*, vol. 16, no. 4, pp. 413–432, 2007.

[56] M. Csikszentmihalyi and I. Csikszentmihalyi, *Optimal Experience: Psychological Studies of Flow in Consciousness*, Cambridge University Press, 2000.

[57] Y. Lu, T. Zhou, and B. Wang, "Exploring Chinese users' acceptance of instant messaging using the theory of planned behavior, the technology acceptance model, and the flow theory," *Computers in Human Behavior*, vol. 25, no. 1, pp. 29–39, 2009.

[58] M. Csikszentmihalyi and J. LeFevre, "Optimal experience in work and leisure," *Journal of Personality and Social Psychology*, vol. 56, no. 5, pp. 815–822, 1989.

[59] J. A. Ghani, "Human factors in information systems," in *Flow in Human-Computer Interactions: Test of a Model*, J. M. Carey, Ed., pp. 291–311, Ablex, 1995.

[60] D. Li and G. J. Browne, "The role of need for cognition and mood in online flow experience," *Journal of Computer Information Systems*, vol. 46, no. 3, pp. 11–17, 2006.

[61] D. Choi and J. Kim, "Why people continue to play online games: in search of critical design factors to increase customer loyalty to online contents," *CyberPsychology & Behavior*, vol. 7, no. 1, pp. 11–24, 2004.

[62] M. C. Ashton, S. V. Paunonen, E. Helmes, and D. N. Jackson, "Kin altruism, reciprocal altruism, and the big five personality factors," *Evolution and Human Behavior*, vol. 19, no. 4, pp. 243–255, 1998.

[63] H. Rachlin, "Altruism and selfishness," *Behavioral and Brain Sciences*, vol. 25, no. 2, pp. 239–250, 2002.

[64] N. Folbre and R. E. Goodin, "Revealing altruism," *Review of Social Economy*, vol. 62, no. 1, pp. 1–25, 2004.

[65] S. H. Burton, R. G. Morris, C. G. Giraud-Carrier, J. H. West, and R. Thackeray, "Mining useful association rules from questionnaire data," *Intelligent Data Analysis*, vol. 18, no. 3, pp. 479–494, 2014.

[66] Y. Hong and Y. Li, "The research on index system optimization of graduation design based on cronbach coefficient," in *Proceedings of the 5th International Conference on Computer Science and Education*, pp. 1843–1845, Hefei, China, August 2010.

[67] B. Williams, A. Onsman, and T. Brown, "Exploratory factor analysis: a five-step guide for novices," *Australasian Journal of Paramedicine*, vol. 8, no. 3, 2010.

[68] C.-L. Hsu and H.-P. Lu, "Consumer behavior in online game communities: a motivational factor perspective," *Computers in Human Behavior*, vol. 23, no. 3, pp. 1642–1659, 2007.

[69] H. van der Heijden, M. Ogertschnig, and L. van der Gaast, "Effects of context relevance and perceived risk on user acceptance of mobile information services," in *Proceedings of the 13th European Conference on Information Systems*, pp. 286–296, 2005.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 4859862, 9 pages
http://dx.doi.org/10.1155/2016/4859862

Research Article

Power-Aware Resource Reconfiguration Using Genetic Algorithm in Cloud Computing

Li Deng,[1,2] Yang Li,[1,2] Li Yao,[1,2] Yu Jin,[1,2] and Jinguang Gu[1,2]

[1] *College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, China*
[2] *Hubei Province Key Laboratory of Intelligent Information Processing and Real-Time Industrial System, Wuhan, China*

Correspondence should be addressed to Li Deng; dengli@wust.edu.cn

Received 23 September 2016; Accepted 12 December 2016

Academic Editor: Qingchen Zhang

Cloud computing enables scalable computation based on virtualization technology. However, current resource reallocation solution seldom considers the stability of virtual machine (VM) placement pattern. Varied workloads of applications would lead to frequent resource reconfiguration requirements due to repeated appearance of hot nodes. In this paper, several algorithms for VM placement (multiobjective genetic algorithm (MOGA), power-aware multiobjective genetic algorithm (pMOGA), and enhanced power-aware multiobjective genetic algorithm (EpMOGA)) are presented to improve stability of VM placement pattern with less migration overhead. The energy consumption is also considered. A type-matching controller is designed to improve evolution process. Nondominated sorting genetic algorithm II (NSGAII) is used to select new generations during evolution process. Our simulation results demonstrate that these algorithms all provide resource reallocation solutions with long stabilization time of nodes. pMOGA and EpMOGA also better balance the relationship of stabilization and energy efficiency by adding number of active nodes as one of optimal objectives. Type-matching controller makes EpMOGA superior to pMOGA.

1. Introduction

Cloud computing [1] provides a huge resource pool shared by a large number of users. Virtualization technology enables dynamic resource configuration according to real demands of applications [2] and live migration of VMs is an important way to implement resource reallocation in the cloud [3].

Wrasse [4] is designed to handle generalized resource allocation in the cloud. It uses massive parallelism by orchestrating a large number of light-weight GPU threads to explore the search space in parallel. Server consolidation [5–7] has always been studied for green computing. Constraint programming is used to reduce the number of active physical nodes for energy efficiency while the Service Level Agreement (SLA) is guaranteed. Efficient VM migration and placement are also helpful for reducing the number of active PMs. Furthermore, economic efficiency of cloud computing has been studied by many researchers [8, 9]. Auction approaches are presented to balance the relationship between economic efficiency and computational efficiency.

However, current resource management methods seldom consider stability of VM placement globally to improve resource efficiency [10]. Due to time-varying resource demands of applications, current mapping of VMs to physical nodes may be not suitable for future workloads. New hot nodes would appear in the near future, which directly results in another resource reallocation. Resource reallocation would subsequently lead to some additional overheads [11], such as migration time, downtime, and service degradation. The stability of a VM placement pattern should be considered during dynamic resource configuration.

Resource allocation problem is a kind of combinatorial problem, known as NP-hard problem [12]. Evolutionary computation algorithm can approximate an optimal solution only taking polynomial time [12]. In this paper, we present several genetic algorithms for resource allocation in cloud computing based on our prior works [10]. According to prediction information of application workloads, these algorithms all provide resource reconfiguration solutions with long stabilization time of nodes. Our contributions are

listed in the following: (1) we design genetic algorithms to better balance the relationship between node stabilization and power efficiency; (2) a type-matching controller is proposed to accelerate evolution process; (3) we implement genetic algorithms and a type-matching controller in Java and compare the performances of these genetic algorithms.

The rest of the paper is organized as follows: Section 2 discusses related work about dynamic resource allocation. In Section 3, we give the description of problem formulation. Objectives and constraints of dynamic resource allocation are formulated. Section 4 introduces the details of several genetic algorithms. Performance evaluation of several algorithms is done in Section 5. Finally, we give our summary and future research directions in Section 6.

2. Related Work

Being completely different from traditional static resource configuration, cloud computing enables dynamic resource allocation based on time-varying workloads of applications. Resource efficiency is thus improved significantly. Many researchers have studied resource reallocation problems.

Dynamic resource allocation usually has the following objectives.

(i) Green Computing. Energy consumption is the most critical problem in cloud computing [13]. It becomes more serious especially in multicore era [14]. Server consolidation [5, 6, 15] is used to decrease the number of active physical nodes. Power efficiency is greatly improved. Constraints programming [5] and genetic algorithm [15] are, respectively, employed to find a solution using the minimum number of active nodes for green computing. An energy-efficient resource allocation framework [7] is proposed to minimize physical node overload occurrences for overcommitted clouds by predicting future resource utilizations of scheduled VMs.

(ii) Resource Fairness. Resource in the cloud is shared among a large number of tenants. Resource fairness among numerous users is then studied [16, 17]. A multiresource allocation mechanism (called `DRFH`) [16] is presented to ensure fair usage of resource among cloud users using heuristics.

(iii) Resource Efficiency. Resource efficiency becomes very important in large-scale datacenters with tens of thousands of servers [18, 19]. Some approaches are designed to improve computing resource utilization, such as memory [20] and I/O [21]. There are some methods presented to improve SLAs of applications [22]. Also, some resource management solutions are proposed for special applications: stream processing [23, 24] and business process [25].

(iv) Economic Efficiency. Resource in the cloud is usually rent in a pay-as-you-go model. Economic efficiency of cloud computing has been studied by many researchers [8, 9]. Trading mechanisms for the demand response are designed to achieve the maximum social welfare with arbitrarily high probability.

In this paper, our work mainly focuses on the stability of VM placement pattern. Because workloads of applications are time-varying especially in mobile cloud computing, the stability becomes more important.

3. Problem Formulation

Due to dynamic workloads, resource demands of applications vary with time. Some nodes have frequent resource contention and become busy when workloads increase. These nodes are called *hot nodes*. Hot nodes should be alleviated by decreasing their workloads to ensure service level objectives (SLAs) of applications.

Live migration of virtual machine is an important method to alleviate hot nodes. It redistributes VMs on a pool of nodes. When remapping VMs to nodes, we should consider future trends of application workloads to avoid "thrashing," much more hot nodes arising in the future. So, stability is an important metric to choose new VM distribution on nodes. The stability of VM distribution mainly depends on the total workloads of each node.

Abbreviations lists the definition of some symbols used in our discussion.

We have the following equations:

$$y_i = \begin{cases} 0, & \text{if } \sum_{j=1}^{\mathcal{N}} x_{ij} = 0, \\ 1, & \text{if } \sum_{j=1}^{\mathcal{N}} x_{ij} \neq 0, \end{cases} \quad i = 1, \ldots, \mathcal{M},$$

$$m_j = \begin{cases} 0, & \text{if } x_{ij} = x'_{i'j} = 1,\ i = i',\ i, i' = 1, \ldots, \mathcal{M}, \\ 1, & \text{if } x_{ij} = x'_{i'j} = 1,\ i \neq i',\ i, i' = 1, \ldots, \mathcal{M}, \end{cases} \quad j = 1, \ldots, \mathcal{N}. \tag{1}$$

Variable x_{ij} denotes node i hosting VM j in old VM placement pattern $\mathcal{D}$, while $x'_{i'j}$ means that VM j resides on node i' in new VM placement pattern $\mathcal{D}'$.

Some definitions are given in Abbreviations.

Definition 1. A *placement pattern* $\mathcal{D}_k$ is the mode in which a group of applications (VMs) are distributed on physical nodes.

Definition 2. The node i is stable if and only if the node has enough resources for applications (VMs) residing on it during a certain period of time, no matter how the workloads of applications vary.

Definition 3. The placement pattern $\mathcal{D}_k$ is stable if and only if each node in the placement pattern is stable during a period of time.

Definition 4. Stabilization time T means the longest period in which a node or a placement pattern stays stable from a certain time. It is a straight-forward metric to measure the stability of a node or a placement pattern. The stabilization

time of a placement pattern depends on that of each node in it, as shown in the following formula:

$$T_{\mathscr{D}_k} = \min\left\{T_{\text{node}_1}, T_{\text{node}_2}, \ldots, T_{\text{node}_{\mathscr{M}}}\right\}. \quad (2)$$

Then, the problem of dynamic resource allocation is formulated as follows: having known dynamic workloads of VMs (including predicted future workloads), given a set of nodes, the objective of dynamic resource allocation is to find a placement solution of VMs on physical nodes with longest stabilization time, minimal number of VM migration, and minimal number of active nodes:

$$\begin{aligned} \text{Objectives:}\quad & \max \quad T_{\mathscr{D}_k}; \\ & \min \quad \sum_{j=1}^{\mathscr{N}} m_j; \\ & \min \quad \sum_{i=1}^{\mathscr{M}} y_i \end{aligned} \quad (3)$$

$$\text{Subject To:}\quad \sum_{i=1}^{\mathscr{M}} x_{ij} = 1, \quad j = 1, \ldots, \mathscr{N} \quad (4)$$

$$\mathscr{C}_i \geq \sum_{j=1}^{\mathscr{N}} x_{ij}\mathscr{C}'_j, \quad i = 1, \ldots, \mathscr{M} \quad (5)$$

$$Mem_i \geq \sum_{j=1}^{\mathscr{N}} x_{ij} Mem'_j, \quad i = 1, \ldots, \mathscr{M} \quad (6)$$

$$x_{ij}, m_j, y_i \in \{0, 1\}, \quad i = 1, \ldots, \mathscr{M},\ j = 1, \ldots, \mathscr{N}. \quad (7)$$

We have three objectives: one is to make the new distribution of VMs with longest stabilization time ($\max T_{\mathscr{D}_k}$); one is to only migrate the minimal number of VMs from current status to new status ($\min \sum_{j=1}^{\mathscr{N}} m_j$); the last one is to use the smallest number of physical nodes. The first objective means that hot nodes would not appear in the new mapping in a short time. The second objective requests that migration overhead of VMs from old status to new status is minimal. The third objective is to make the number of active physical nodes as small as possible for energy efficiency.

In the above formulae, formula (4) indicates that each VM only resides on one physical node. Formula (5) means that the total amount of CPU resource requested by VMs residing on the same node is not larger than the amount of resource supplied by the node. Formula (6) denotes that the total amount of memory requested by VMs is not larger than the amount of memory supplied by the node. Formula (7) explains that x_{ij}, m_j, and y_i are binary variables.

4. Resource Reconfiguration Approach

As dynamic resource allocation problem is a kind of NP-complete problem, it is hard to find the optimal solution in polynomial time. Using evolution theory of biosphere, genetic algorithm can find an approximately optimal solution to resource allocation problem through simulating biologic evolution process.

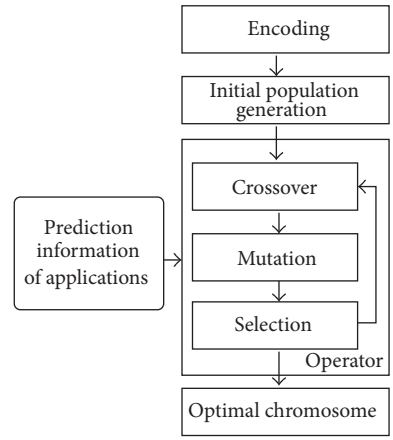
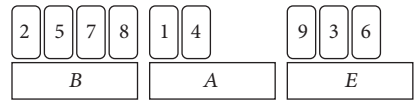


Figure 1: Flow chart of genetic algorithm.



Chromosome: *BEA* (*B*{2, 5, 7, 8}, *E*{1, 4}, *A*{9, 3, 6})

Figure 2: Examples of group encoding scheme.

We propose three algorithms: multiobjective genetic algorithm (MOGA), power-aware multiobjective genetic algorithm (pMOGA), and enhanced power-aware multiobjective genetic algorithm (EpMOGA). MOGA only aims at two objectives: long stability time of VM distribution and minimal number of VM migration. Different from MOGA, pMOGA adds a new objective to be optimized for energy efficiency, shown as formula (3). EpMOGA introduces a type-matching controller based on pMOGA. The type-matching controller is designed to speed up evolution process by matching the type of genes.

4.1. Key Parts of Genetic Algorithm. There are several key parts in genetic algorithm: encoding, initial population generation, main operators (crossover, mutation, and selection), and termination condition, as shown in Figure 1. MOGA, pMOGA, and EpMOGA have the same encoding, the same initial population generation, and the same termination condition.

Encoding. Encoding is to express chromosomes, genes with elements of resource allocation problem. There are three methods to express bin packing problems in genetic algorithm: one gene per object, one gene per bin, and one gene per group (bin and objects in it) [26]. The encoding scheme based on group is employed because it can exactly express the relationship between VMs and physical nodes.

Figure 2 lists examples of encoding scheme using group. In Figure 2, nine VMs are deployed on three nodes. Accordingly, there are three genes in the form of chromosome. Each

gene includes one physical node and several VMs residing on it. A chromosome or an individual signifies a possible solution, a mapping between virtual machines and physical nodes.

Initial Population Generation. A population is a set of chromosomes. Let the population size be *popSIZE*. Genetic algorithm usually starts from an initial population which is often generated randomly. Random generation provides wide search space to find a solution, but it takes much time to get an optimal global solution. First-fit heuristic is used to generate the first population. Note that each individual should meet the constraints discussed in Section 3.

Termination Condition. We set value of the maximum generation (*MAX_GEN*). Iterations would stop when the maximum generation (*MAX_GEN*) is reached.

The difference of the three algorithms mainly lies in operator crossover, mutation, and selection. The difference is discussed below.

4.2. Multiobjective Genetic Algorithm (MOGA). Multiobjective genetic algorithm only has two objectives: long stability time of VM placement and small number of VM migration.

Three main operators (crossover, mutation, and selection) in genetic algorithm are discussed in the following.

Crossover. Crossover is for two parents to produce offspring so that children can inherit much of meaningful information from parents. Using group encoding scheme, chromosomes may have different length. Crossover should be done on chromosomes with varied length.

There are mainly four steps in operator crossover:

(1) Two chromosomes are randomly selected as parents and crossing sites on each parent are chosen at random in both parents.

For example, chromosome $BEA(B\{2,5,7,8\}, E\{1,4\}, A\{9,3,6\})$ and $CBED(C\{5,9\}, B\{2\}, E\{1,6,7\}, D\{4,3,8\})$ are selected as parents. Genes $A\{9,3,6\}$ and $B\{2\}$ are, respectively, crossing sites.

(2) Two parent chromosomes exchange genes at crossing sites.

After exchanging genes, the above two chromosomes become $BEB(B\{2,5,7,8\}, E\{1,4\}, B\{2\})$ and $CAED(C\{5,9\}, A\{9,3,6\}, E\{1,6,7\}, D\{4,3,8\})$.

(3) Some genes with repeated nodes or VMs should be removed. So, the above chromosomes change to $EB(E\{1,4\}, B\{2\})$ and $A(A\{9,3,6\})$.

(4) Some missing VMs are reinserted into genes using first fit decreasing (FFD) heuristic.

In the above example, the missing VMs of the first chromosome include VMs 3, 5, 6, 7, 8, and 9. These VMs should be located on active nodes again. If active nodes do not have enough resource to host these missing VMs, idle nodes are activated.

Crossover operator is done by rate q_c. A population generation produces offsprings with the same size as parents.

Mutation. Mutation may make an individual in the population different from his parents. It adds new information in an arbitrary way to widen search space and avoids being trapped at local optima.

Given a small mutation rate q_m, some chromosomes in the population are selected randomly to execute operator mutation. Mutation is to delete some genes at random in chromosomes. The missing VMs should be relocated to other nodes using FFD.

Selection. Operator selection is to select the new population generation from the old generation and their offsprings. A fast multiobjective genetic algorithm (NSGA-II) [27] is used for operator selection. NSGA-II suits well for constrained multiobjective optimization in any evolutionary algorithm [27].

Each chromosome l has two attributes: nondomination rank (l_{rank}) and crowding distance (l_{distance}) [27]. The smaller the nondomination rank is, the closer the chromosome is to the optimal solution. In the same nondomination rank, the bigger the crowding distance is, the better the chromosome is.

MOGA aims at a resource reconfiguration solution with long stability time of VM placement and small number of VM migrations. Relationship *dominate* between two chromosomes (l, k) is defined as follows:

$$l \textit{ dominate } k, \quad \text{iff } \mathcal{T}_l > \mathcal{T}_k,\ Y_l < Y_k. \tag{8}$$

$\mathcal{T}_l$, $\mathcal{T}_k$ means the stability time of chromosome l, k and Y_l, Y_k denotes the number of VM migration, respectively. Then, we have the following equations (S denotes the set of chromosomes):

$$\begin{aligned} &l_{\text{rank}} = 1, \quad \text{if } \neg\left(\exists k \in S \wedge k \textit{ dominate } l\right). \\ &l_{\text{rank}=k_{\text{rank}}+1}, \quad \text{if } \left(\left(\exists k \in S \wedge k \textit{ dominate } l\right), \text{ for } \left(\forall u \in S \wedge u \textit{ dominate } l\right), \text{ having } k_{\text{rank}} \leq u_{\text{rank}}\right). \end{aligned} \tag{9}$$

The crowding distance is computed as the sum of each normalized objective function [27]. A partial order $\prec$ between two chromosomes l and k is defined. Let $l \prec k$, if ($l_{\text{rank}} < k_{\text{rank}}$ or (($l_{\text{rank}} = k_{\text{rank}}$) and ($l_{\text{distance}} > k_{\text{distance}}$))). Apparently, poset $(S, \prec)$ (S denotes a set of chromosomes in a population generation) is also a well-ordered set. S is a totally ordered set. Chromosomes in set S can be ordered into a chain according to total order $\prec$.

When *popSIZE* parent chromosomes produce *popSIZE* offsprings, all these chromosomes form a big set S' with

($2 * popSIZE$) elements together. Then, selection operator chooses the first $popSIZE$ chromosomes as a new generation from set S' based on total order $\prec$.

4.3. Power-Aware Multiobjective Genetic Algorithm (pMOGA). Power-aware multiobjective genetic algorithm takes power efficiency into consideration based on MOGA. Optimized objectives are listed in formula (3).

Operator crossover and mutation in pMOGA are the same as those in MOGA. Operator selection is discussed below.

Operator selection is still based on NSGA-II. Each chromosome l has two attributes: nondomination rank (l_{rank}) and crowding distance (l_{distance}). The computation of two attributes is like the computation in MOGA. Only crowding distance is computed as the sum of three normalized objective functions in pMOGA, while it is figured out based on two normalized objective functions in MOGA.

In pMOGA, relationship *dominate* between two chromosomes (l, k) is defined in the following:

$$l \text{ dominate } k, \quad \text{iff } \mathcal{T}_l > \mathcal{T}_k,\ Y_l < Y_k,\ A_l < A_k. \tag{10}$$

$\mathcal{T}_l$, $\mathcal{T}_k$ means the stability time of chromosome l, k and Y_l, Y_k denotes the number of VM migration, respectively. Variables A_l and A_k express the number of active physical nodes in chromosome l, k.

4.4. Enhanced Power-Aware Multiobjective Genetic Algorithm (EpMOGA). Enhanced power-aware multiobjective genetic algorithm (EpMOGA) is designed to add a type-matching controller to pMOGA. The controller is mainly used in operator crossover and mutation. EpMOGA and pMOGA have the same operator selection.

As shown in Figures 3 and 4, when placing missing VMs, pMOGA uses FFD and EpMOGA employs a type-matching controller, which is the only difference between pMOGA and EpMOGA.

In cloud computing, the workloads of various applications are multiattribute in terms of different types of resources (CPU, memory, etc.) [28]. A type-matching controller is thus designed to classify applications and nodes into several categories and match them effectively. According to workloads of applications, VMs are classified into CPU-intensive (CI), memory-intensive (MI), both of CPU-intensive and memory-intensive (CMI), none of CPU-intensive and memory-intensive (Non). The type of a VM usually keeps unchanged during their whole lifetime. Also, active physical nodes are sorted into the same four classes. But the type of an active node would vary when it hosts different VMs.

In our experiments, we find that when the same VM migrates to different types of nodes, these nodes have diverse stabilization time. So, we define closeness degree of each type of active nodes for every class of VMs, which is listed in Table 1. As shown in Table 1, the smaller the value of type closeness degree is, the longer the stabilization time of nodes hosting VMs is. When selecting a destination node for a VM, the type-matching controller first tries to match VM to nodes with low type closeness degree. Only when there is not any node with low type closeness degree available are nodes with high closeness degree considered as candidates.

Table 1: Type closeness degree of active nodes to VMs.

Type of VMs	Type of active nodes			
	CI	MI	CMI	Non
CI	4	1	3	2
MI	1	4	3	2
CMI	3	2	4	1
Non	2	3	1	4

When placing missing VMs, type-matching controller tries to map VMs to nodes with appropriate type. It can avoid resource contention and improve resource utilization effectively at the same time to place a CPU-intensive VM on a memory-intensive node. For a CPU-intensive VM, if there is not any memory-intensive active node available, type-matching controller would try to find a node with type Non. If there is not any node with type Non available, a CPU-intensive node is then sought.

5. Performance Evaluation

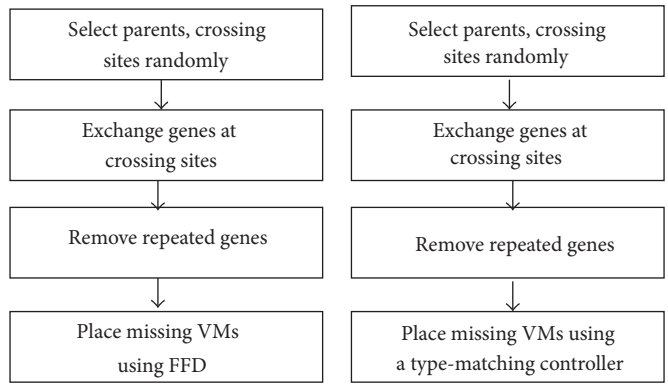
In this section, we evaluate the performance of MOGA, pMOGA, and EpMOGA. All the above algorithms are coded in Java and CloudSim [29] is used to simulate a cloud computing infrastructure. Our tests are done on a ASUS K46CM with Intel Core i5 CPU, 4GB RAM, and 1TB hard drive.

We simulate 58 physical nodes and 174 VMs. Resource requests (only CPU and memory) of these VMs are randomly generated as prediction information. Population size is set as 32 ($popSIZE = 32$). The value of constant MAX_GEN, the maximum generation to produce in genetic algorithms, is set as 40 ($MAX\_GEN = 40$). Crossover rate (q_c) is 0.7 and mutation rate (q_m) is 0.05.

5.1. Evolutionary Process of EpMOGA. Convergence and stability of algorithms are first checked. We observe the evolution process of EpMOGA from the 8th population to the maximum generation.

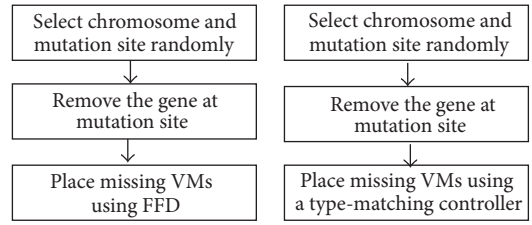
Figure 5 depicts the evolutional process of EpMOGA. x-axis expresses number of VM migrations of each chromosome. y-axis shows stabilization time in seconds. z-axis depicts number of active nodes. Number of VM migrations is just estimated roughly by comparing source node and destination node of each VM. Only five generations (the 8th, 16th, 24th, 32th, and 40th generation) are listed in the figure. Each generation has 32 chromosomes.

From Figure 5, we can find that the reproduction process of individuals moves gradually towards the best solution (longer stabilization time, less number of VM migrations, and less number of active nodes). The process begins with quick changes. The 8th population is quite different from the 16th generation. But the change becomes small in the latter. The 32nd generation is close to the 40th generation. Figure 5 shows that the 40th generation is enough to find the best solution of VM placement in cloud computing.



(a) Crossover in pMOGA (b) Crossover in EpMOGA

Figure 3: Contrast between operator crossovers in pMOGA and EpMOGA.



(a) Mutation in pMOGA (b) Mutation in EpMOGA

Figure 4: Contrast between operator mutations in pMOGA and EpMOGA.

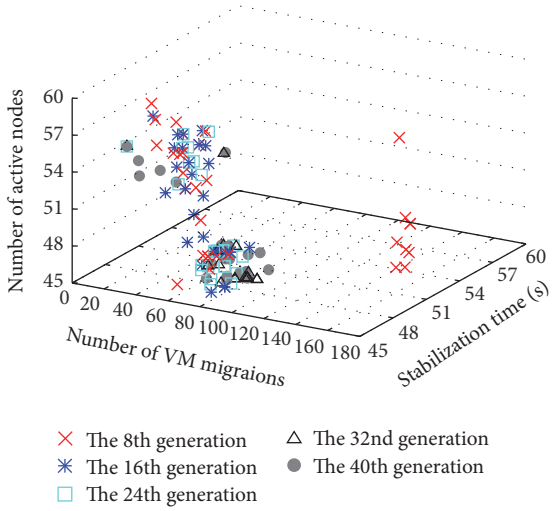


Figure 5: Evolutional process of EpMOGA.

5.2. Comparison of MOGA, pMOGA, and EpMOGA. In this part, we compare the performances of MOGA, pMOGA, and EpMOGA. In environment with the same initial VM placement and the same resource prediction information, MOGA, pMOGA, and EpMOGA, respectively, find a new VM placement. We compare their stabilization time, number of active nodes, and redistribution overhead (denoted as number of VM migrations). Average power $\overline{p}$ is roughly computed using formulae (11).

In formulae (11), $T_{\mathscr{D}}$ denotes stability time of a VM placement pattern $\mathscr{D}$. E_{node} means energy consumed by all active physical nodes ($\sum_{i=1}^{M} y_i$) in pattern $\mathscr{D}$. $\overline{P_{\text{server}}}$ denotes average power of servers. Here, $\overline{P_{\text{server}}}$ is set as 400 watts [7]. E_{mig} expresses energy consumed during VM migration, which is only related to network traffic in migration process [30]. Network traffic is mainly based on the amount of memory of migrated VMs (expressed as $\sum_j Mem'_j$). Parameters k_1, k_2, and k_3 are, respectively, set as 0.512, 1.5, and 20.165, which are got by training models [30].

$$\begin{aligned}\overline{p} &= \frac{\left(E_{\text{node}} + E_{\text{mig}}\right)}{t} \\ &= \frac{\left(T_{\mathscr{D}} * \sum_{i=1}^{M} y_i * \overline{P_{\text{server}}} + \left(k_1 * k_2 * \sum_j Mem'_j + k_3\right)\right)}{T_{\mathscr{D}}}.\end{aligned} \tag{11}$$

We normalize performance values of pMOGA and EpMOGA after setting all the performance values of MOGA as 1. The results are listed in Figure 6. From Figure 6, we find that both pMOGA and EpMOGA have less number of active nodes and less average power at the cost of shorter stability time and larger number of VM migrations. With a type-matching controller, EpMOGA has better performance values than pMOGA. Average power of EpMOGA is 0.818 times that of MOGA and 0.922 times the power of pMOGA.

Figure 6 shows that MOGA has the longest stabilization time and the smallest number of VM migrations. But pMOGA and EpMOGA better balance the relationship

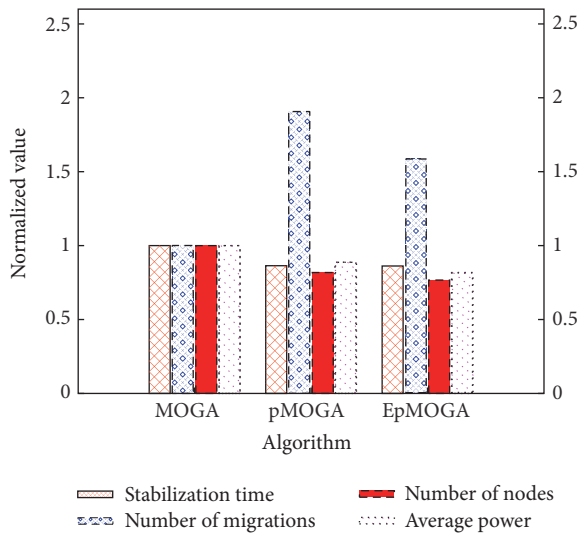


Figure 6: Performance comparison of several algorithms.

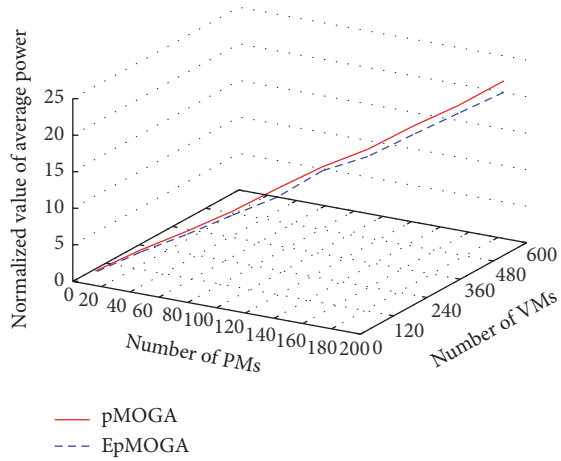


Figure 7: Average power of pMOGA and EpMOGA.

of VM distribution stabilization and power efficiency by adding number of active nodes as one of optimization objectives. Number of active nodes is one of the main power consumption factors in cloud computing. pMOGA and EpMOGA migrate more VMs to use less active nodes, saving more energy consumption. With a type-matching controller, EpMOGA has better solution than pMOGA. The controller helps to optimize evolution process for optimal objectives.

We change number of nodes and number of VMs to test average power of pMOGA and EpMOGA. We set the minimum power in test results as 1 and normalize other power values. The test results are shown in Figure 7. With the increase of VMs and PMs, average power of pMOGA and EpMOGA rises up. EpMOGA always finds a solution with less power than pMOGA. The more VMs and PMs there are, the clearer advantage EpMOGA has. Figure 7 demonstrates that the type-matching controller is helpful to accelerate evolution process for optimal objectives.

6. Conclusion and Future Work

In this paper, several genetic algorithms have been proposed to implement dynamic resource allocation for stability in cloud computing. The group encoding scheme is employed to clearly express the mapping of VMs and physical nodes. A type-matching controller is designed to speed up evolution process. Our simulation results show that these genetic algorithms effectively improve stability of VM redistribution. Also, pMOGA and EpMOGA both better balance the relationship of stabilization and energy efficiency. With type-matching controller, EpMOGA is superior to pMOGA.

In the future, we will continue to work on dynamic resource configuration in cloud computing using genetic algorithms. We find that when there are more objectives to be optimized, nondominated sorting genetic algorithm II is less effective. Many chromosomes are in the same nondomination rank. A new sorting algorithm should be studied.

Abbreviations

$\mathscr{M}$: The total number of physical nodes in the cloud
$\mathscr{N}$: The total number of virtual machines in the cloud
$\mathscr{C}_i$: The amount of CPU resource that node i supplies
Mem_i: The amount of memory resource that node i supplies
$\mathscr{C}'_j$: The amount of CPU resource that VM j requests
Mem'_j: The amount of memory resource that VM j requests
x_{ij}: Binary variable; if $x_{ij} = 1$, node i hosts VM j, or else, $x_{ij} = 0$
y_i: Binary variable; $y_i = 1$ if node i is active and hosts one VM at least, or else, $y_i = 0$
$\mathscr{D}_k$: The kth placement pattern of all VMs in the cloud
m_j: Binary variable; if $m_j = 1$, VM j migrates once, or else, $m_j = 0$
T_{node_i}: Stabilization time of a node, node_i
$T_{\mathscr{D}_k}$: Stabilization time of a placement pattern $\mathscr{D}_k$.

Competing Interests

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

Acknowledgments

This research was supported by the Opening Project of Hubei Key Laboratory of Intelligent Information Processing and Real-Time Industrial System in China (no. 2016znss27B), the National Nature Science Foundation of China (no. 61303117 and no. 61272110), and the Key Projects of National Social Science Foundation of China under Grant no. 11&ZD189.

References

[1] M. Armbrust, A. Fox, R. Griffith et al., "Above the clouds: a Berkeley view of cloud computing," Tech. Rep. UCB/EECS-2009-28, Electrical Engineering and Computer Sciences Department, University of California, Berkeley, 2009.

[2] G. Copil, D. Moldovan, H. Truong, and S. Dustdar, "rSYBL: a framework for specifying and controlling cloud services elasticity," *ACM Transactions on Internet Technology*, vol. 16, no. 3, 2016.

[3] H. Jin, L. Deng, S. Wu, X. H. Shi, H. H. Chen, and X. D. Pan, "MECOM: live migration of virtual machines by adaptively compressing memory pages," *Future Generation Computer Systems*, vol. 38, pp. 23–35, 2014.

[4] A. Rai, R. Bhagwan, and S. Guha, "Generalized resource allocation for the cloud," in *Proceedings of the ACM 3rd Symposium on Cloud Computing (SOCC '12)*, San Jose, Calif, USA, 2012.

[5] F. Hermenier, X. Lorca, J. M. Menaud, G. Muller, and J. Lawall, "Entropy: a consolidation manager for clusters," in *Proceedings of the ACM/Usenix International Conference on Virtual Execution Environments (VEE '09)*, pp. 41–50, Washington, DC, USA, March 2009.

[6] L. Chen and H. Shen, "Consolidating complementary VMs with spatial/temporal-awareness in cloud datacenters," in *Proceedings of the 33rd IEEE Conference on Computer Communications (INFOCOM '14)*, pp. 1033–1041, IEEE, Toronto, Canada, May 2014.

[7] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "An energy-efficient VM prediction and migration framework for overcommitted clouds," *IEEE Transactions on Cloud Computing*, 2016.

[8] L. Zhang, Z. Li, and C. Wu, "Dynamic resource provisioning in cloud computing: a randomized auction approach," in *Proceedings of the 33rd IEEE Conference on Computer Communications ('INFOCOM '14)*, pp. 433–441, Ontario, Canada, May 2014.

[9] Z. Zhou, F. Liu, Z. Li, and H. Jin, "When smart grid meets geo-distributed cloud: an auction approach to datacenter demand response," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 2650–2658, IEEE, May 2015.

[10] L. Deng and L. Yao, "Dynamic allocation of virtual resources based on genetic algorithm in the cloud," in *Proceedings of the Asia-Pacific Services Computing Conference (APSCC '15)*, pp. 153–164, 2015.

[11] S. Nathan, U. Bellur, and P. Kulkarni, "Towards a comprehensive performance model of virtual machine live migration," in *Proceedings of the 6th ACM Symposium on Cloud Computing (SoCC '15)*, pp. 288–301, Kohala Coast, Hawaii, USA, August 2015.

[12] Z.-H. Zhan, X.-F. Liu, Y.-J. Gong, J. Zhang, H. S.-H. Chung, and Y. Li, "Cloud computing resource scheduling and a survey of its evolutionary approaches," *ACM Computing Surveys*, vol. 47, no. 4, article 63, 2015.

[13] T. Kaur and I. Chana, "Energy efficiency techniques in cloud computing: a survey and taxonomy," *ACM Computing Surveys*, vol. 48, no. 2, article 22, 2015.

[14] J. Liu, K. L. Li, D. K. Zhu, J. J. Han, and K. Q. Li, "Minimizing cost of scheduling tasks on heterogeneous multicore embedded systems," *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 2, 2016.

[15] Q. Li, Q.-F. Hao, L.-M. Xiao, and Z.-J. Li, "Adaptive management and multi-objective optimization for virtual machine placement in cloud computing," *Chinese Journal of Computer*, vol. 34, no. 12, pp. 2253–2264, 2011.

[16] W. Wang, B. Li, and B. Liang, "Dominant resource fairness in cloud computing systems with heterogeneous servers," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '14)*, pp. 583–591, IEEE, Toronto, Canada, May 2014.

[17] J. Guo, F. Liu, J. C. S. Lui, and H. Jin, "Fair network bandwidth allocation in IaaS datacenters via a cooperative game approach," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 873–886, 2016.

[18] D. Lo, L. Cheng, R. Govindaraju, P. Ranganathan, and C. Kozyrakis, "Improving resource efficiency at scale with heracles," *ACM Transactions on Computer Systems*, vol. 34, no. 2, 2016.

[19] S. Singh and I. Chana, "QoS-aware autonomic resource management in cloud computing: a systematic review," *ACM Computing Surveys*, vol. 48, no. 3, article 42, 2016.

[20] K. H. Park, W. Hwang, H. Seok et al., "MN-MATE: elastic resource management of manycores and a hybrid memory hierarchy for a cloud node," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 12, no. 1, article 5, 2015.

[21] R. C. Chiang, S. Rajasekaran, N. Zhang, and H. H. Huang, "Swiper: exploiting virtual machine vulnerability in third-party clouds with competition for I/O resources," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 6, pp. 1732–1742, 2015.

[22] N. Jain, I. Menache, J. Naor, and J. Yaniv, "Near-optimal scheduling mechanisms for deadline-sensitive jobs in large computing clusters," *ACM Transactions on Parallel Computing*, vol. 2, no. 1, 2015.

[23] J. Ghaderi, S. Shakkottai, and R. Srikant, "Scheduling storms and streams in the cloud," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, vol. 1, no. 4, 2016.

[24] T. Wu, W. Dou, F. Wu, S. Tang, C. Hu, and J. Chen, "A deployment optimization scheme over multimedia big data for large-scale media streaming application," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 5, article 73, 2016.

[25] J. Xu, C. Liu, X. Zhao, S. Yongchareon, and Z. Ding, "Resource management for business process scheduling in the presence of availability constraints," *ACM Transactions on Management Information Systems*, vol. 7, no. 3, article 9, 2016.

[26] E. Falkenauer and A. Delchambre, "A genetic algorithm for bin packing and line balancing," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1186–1192, Nice, France, May 1992.

[27] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[28] L. Chen, H. Shen, and K. Sapra, "Distributed autonomous virtual resource management in datacenters using finite-Markov decision process," in *Proceedings of the 5th ACM Symposium on Cloud Computing (SOCC '14)*, pp. 1–13, ACM, Seattle, Wash, USA, November 2014.

[29] CloudSim: A Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services, 2015, http://www.cloudbus.org/cloudsim/.

[30] H. Liu, C.-Z. Xu, H. Jin, J. Gong, and X. Liao, "Performance and energy modeling for live migration of virtual machines," in *Proceedings of the 20th ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC '11)*, pp. 171–181, ACM, San Jose, Calif, USA, June 2011.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 2436074, 18 pages
http://dx.doi.org/10.1155/2016/2436074

Research Article

Delay-Aware Program Codes Dissemination Scheme in Internet of Everything

Yixuan Xu,[1] Anfeng Liu,[1] and Changqin Huang[2,3]

[1]*School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2]*School of Information Technology in Education, South China Normal University, Guangzhou 510631, China*
[3]*Beijing Hetian Yuxiang Internet Technology Co., Ltd., Beijing 100036, China*

Correspondence should be addressed to Anfeng Liu; afengliu@mail.csu.edu.cn

Received 21 September 2016; Revised 5 November 2016; Accepted 16 November 2016

Academic Editor: Qingchen Zhang

Due to recent advancements in big data, connection technologies, and smart devices, our environment is transforming into an "Internet of Everything" (IoE) environment. These smart devices can obtain new or special functions by reprogramming: upgrade their soft systems through receiving new version of program codes. However, bulk codes dissemination suffers from large delay, energy consumption, and number of retransmissions because of the unreliability of wireless links. In this paper, a delay-aware program dissemination (DAPD) scheme is proposed to disseminate program codes with fast, reliable, and energy-efficient style. We observe that although total energy is limited in wireless sensor network, there exists residual energy in nodes deployed far from the base station. Therefore, DAPD scheme improves the performance of bulk codes dissemination through the following two aspects. (1) Due to the fact that a high transmitting power can significantly improve the quality of wireless links, transmitting power of sensors with more residual energy is enhanced to improve link quality. (2) Due to the fact that performance of correlated dissemination tends to degrade in a highly dynamic environment, link correlation is autonomously updated in DAPD during codes dissemination to maintain improvements brought by correlated dissemination. Theoretical analysis and experimental results show that, compared with previous work, DAPD scheme improves the dissemination performance in terms of completion time, transmission cost, and the efficiency of energy utilization.

1. Introduction

Due to recent advancements in big data, connection technologies, and smart devices, the number of connected devices has already exceeded the number of people on Earth since 2011. Connected smart devices have reached 9 billion and are expected to grow more rapidly and reach 24 billion by 2020 [1]. Our environment is transforming into an "Internet of Everything" (IoE) environment. In 2012, global commercialization of IoT-based application systems generated a revenue of $4.8 trillion [2]. Cisco estimates that, due to IoT, the global corporate profits will also increase approximately by 21% [3]. Due to extremely low costs of sensors and actuators, they can surely find their places in a wide range of applications in smart factory, smart city, and smart life, which lead to "Internet of Everything" (IoE) [4–8].

For example, many smart wireless sensors have been deployed in smart factory to monitor states of machines, sensing temperature, humidity, and sound [9, 10]. Wireless sensors are well suited for complicated industry environment because the deployment of them requires no wiring, so they have already been widely used in industrial production fields. Smart factory, which is composed of smart wireless sensors, can collect various kinds of data from machines and mine these collected data (i.e., industrial big data) to obtain valuable information for factory operation [11]. Machines are automatically controlled by obtained information to make an efficient production line (i.e., adequate production speed, low power consumption, and failure prediction). Therefore, smart wireless sensors make it possible to optimize the factory operation without human resources.

A sensor will work for several months or years once it is deployed [12]. However, in order to gain new functions, the upgrade of industrial production line requires sensors to upgrade simultaneously. One method called reprogramming is considered to be economic and convenient for such operations [13–16]. Besides, sometimes even without the upgrade of manufacturing facilities, these sensors will also need to upgrade to adapt to changes on production requirements. Therefore, in "Internet of Everything" (IoE) environment (e.g., smart factory, smart city, and smart life), it is common to disseminate new program codes to all wireless sensors through wireless communication. In this paper, such operation is called codes dissemination. Codes dissemination is a significant and crucial technique when sensors are deployed in environments where physically operating and reprogramming them are difficult or unfeasible. As a basic operation to enable wireless reprogramming, it attracts many research attentions in recent years. However, codes dissemination faces many challenges. First, the length of program codes is longer than the length of code packets and a network may include thousands of sensor nodes. Thus, disseminating large size codes correctly to a great amount of sensor nodes is one challenging issue. Another issue is dissemination delay (i.e., dissemination completion time, DCT), which refers to the required time for disseminating codes to all sensor nodes. It is better to obtain less dissemination completion time because large dissemination completion time may cause codes of different nodes to be inconsistent, resulting in loss of application due to the chaos about communication and signal transmission. Third, it is important to ensure complete reliability, which means that each active node in the network should receive program codes completely and correctly. Thus, large-scale programming codes dissemination for "Internet of Everything" (IoE) environment is a challenging task.

There are mainly three kinds of codes dissemination schemes. (1) The first one is a scheme called deluge [17]; this scheme uses negotiation to improve the performance of reliability. The method used in this scheme can be divided into three stages: broadcast, request, and send. Due to the fact that it needs three operations for each transmission, this scheme costs much to ensure the reliability and transmission delay. (2) The second scheme is flooding-based dissemination scheme [18]; this scheme removes request stage, which can make the speed for spreading program codes faster. However, the disadvantage is to cause broadcast storm problem. (3) The third scheme is link correlation-aware data dissemination scheme, which is proposed in [19]. The main idea of correlated dissemination (CD) is disseminating codes to the whole network by the broadcasting of sensor nodes; thus it is a one-to-many operation in unreliable wireless networks. Nodes which start broadcasting are called parent nodes and nodes which receive codes are called child nodes. In CD scheme, each node can only choose one node as its parent node and this parent node will take the responsibility of broadcasting codes to all its child nodes. Link correlation refers to the proportion of packets that are successfully (or unsuccessfully) received by all child nodes during broadcasting. Assigning sensor nodes with high link correlation to a same parent node can make retransmission packets more likely to be needed by more than one child node and therefore number of retransmissions, dissemination completion time, and energy consumption are reduced. The main innovative idea of CD scheme is successfully building up a model to estimate link correlation and reducing number of retransmissions according to it.

Although many researches have already been done, some problems deserve further study [20]. (1) The first problem is the problem on improving the reliability of wireless link. Previous researches normally ensure the reliability of codes dissemination through multiple retransmissions in network layer. However, such solutions also bring problems on increasing the delay of codes dissemination and energy consumption, especially in a network with high packet loss ratio. Therefore, how to ensure link reliability and maintain low delay simultaneously is an important and challenging issue. (2) Although correlated dissemination (CD) is able to reduce retransmission packets, it faces the problem of sending extra packets to obtain link correlation, which consumes more energy of sensors and shortens the lifetime of network. Besides, link correlations tend to change dynamically during real-world codes dissemination. Therefore, how to overcome the extra energy consumption of sensors and obtain the latest link correlation simultaneously is another big concern.

Based on the analysis above, a delay-aware program dissemination (DAPD) scheme is proposed to disseminate program codes with fast, reliable, and energy-efficient style. The improvement of DAPD on the performance of codes dissemination is founded on two facts in wireless sensor network. (1) Link quality is related to transmitting power of sensor nodes directly. The former one will improve greatly when enhancing transmitting power, which leads to a decrease in the number of retransmissions and DCT. Besides, energy consumption on retransmitting codes also reduces. On the other hand, although total energy of sensors is limited, sensors are actually in a state of sensing machines during most of time, transmitting collected data to the base station by multihop. With this many-to-one operation, sensors around the base station not only need to transmit their own data but also take the responsibility of forwarding data originated from sensors far from the base station (far nodes). Therefore, energy consumption of these sensors is much larger than far nodes and residual energy will accumulate in far nodes during this period. If such residual energy can be exploited when disseminating codes, the performance will improve greatly. (2) The premise for correlated dissemination is using extra packets to obtain link correlation before disseminating codes. If the number of these packets is too small, link correlation cannot be obtained correctly and adequately. On the other hand, if it is too large, the lifetime of network will be shortened due to much extra energy consumption. So sensor nodes far from the base station (also with excess energy) are able to obtain link correlations adequately with our scheme to achieve the goal that the lifetime of network will not be influenced while dissemination performance is improved. The main contributions of the DAPD scheme are listed as follows.

(1) DAPD scheme enhances transmitting power of sensors with excess energy to improve link quality and reduce

dissemination completion time under the premise of not shortening the lifetime of network.

(2) DAPD scheme takes full advantage of residual energy to obtain link correlations correctly and adequately before codes dissemination. Besides, it also proposes a dynamic parent node selection algorithm during codes dissemination to quickly adapt to the latest environment and make use of link correlation more effectively.

(3) Through our theoretical analysis and simulation study, we demonstrate that, for DAPD scheme, codes dissemination completion time can be reduced and energy utilization efficiency can be enhanced simultaneously. Compared with former schemes, codes dissemination completion time can be reduced by as much as 19.05% (larger when the environment is highly dynamic). More importantly, the proposed scheme improves the performances without harming network lifetime, which is difficult to achieve in previous schemes.

The rest of this paper is organized as follows: Section 2 reviews related work. System models and problem statements are introduced in Section 3. In Section 4, a novel DAPD scheme is presented to disseminate program codes with fast, reliable and energy-efficient style. Performance analysis for DAPD scheme is provided in Section 5. Experimental results and comparisons are given in Section 6. Section 7 concludes the paper.

2. Related Work

Many program codes dissemination schemes have been proposed [21–28], with each of them focusing on one or two specific challenges during codes dissemination phase (e.g., latency, energy consumption, and reliability). Generally, these schemes can be divided into the following types based on their design purposes and requirements.

(1) The first type is schemes focusing on reducing latency. The objective of such schemes is to ensure the reliability of codes dissemination and reduce dissemination completion time simultaneously. Deluge can be considered as an example of such schemes [17]. Three-way handshake and ACK-based protocol are adopted in deluge for reliability. Besides, transmission delay can be improved through dividing codes into fixed size pages. In deluge, each node will advertise about local pages. When one node (receiver) learns that another node (sender) has pages not successfully received by itself, it will send a request to the sender and prepare to receive pages. Many other schemes are based on deluge. For example, rateless deluge reduces latency further through using random linear codes to encode packets [21].

Zheng et al. proposed Survival of the Fittest (SurF) to solve the problem that negotiations between sensor nodes tend to incur long dissemination completion time [23]. This scheme achieves a tradeoff between negotiation and flooding, which is another dissemination scheme but is considered to be energy-consuming. SurF selectively adopts two schemes (negotiation and flooding) to reduce dissemination completion time.

(2) The second type is schemes focusing on reducing energy consumption. The objective of such schemes is to prolong the lifetime of wireless sensor network. In many energy-efficient schemes, sensor nodes alternate between active state and dormant state to reduce energy consumption [24, 25, 28, 29]. Kulkarni and Wang proposed one scheme called MNP after finding that one main source of energy consumption in deluge results from high degree of message collision [26]. In MNP, a sender selection algorithm is used to solve message collision problem. Besides, one node will go into dormant state if its neighbor nodes are transmitting packets already owned by itself. Due to a decrease on active radio time, energy consumption is significantly reduced. In addition, experiences show that single-hop reprogramming may achieve a better performance on dissemination completion time and energy consumption than multihop reprogramming under certain conditions. Therefore, one scheme called DStream is proposed [27], which has the abilities on both single-hop and multihop dissemination.

(3) Another kind of special dissemination scheme proposed recently is called link correlation-aware data dissemination scheme [19]. Due to the fact that codes dissemination is one kind of wireless broadcast in this scheme, disseminated codes will be received by more than one sensor in the broadcast domain. Basically, sensors with high link correlation are more likely to successfully (or unsuccessfully) receive packets with same packet ID. On the other hand, lost packets of sensors with low link correlation tend to be different from each other. Therefore, this scheme combines sensors with high correlation together, which makes retransmission packets have more possibilities to be needed by more than one sensor. Correlated dissemination performs well in terms of latency, energy consumption, and reliability when the environment of network is stable.

3. System Models and Problem Statements

3.1. Network Model. The network model that we adopt is shown in Figure 1, which can also be found in [9]. The whole network is composed of one base station and many sensor nodes evenly distributed in the network. The energy of base station is considered to be infinite. In contrast, sensor nodes are powered by batteries and total energy is limited.

Two main functions of network include data collection and codes dissemination. During the first operation, all sensor nodes need to transmit collected data back to base station by multihop with unicast style. Due to the unreliability of link, sensors may have to transmit data repeatedly. Packets transmitted during the first operation are called data packets in our scheme. The base station will transmit code packets to all sensor nodes in the second phase and sensor nodes will also participate in transmitting these code packets through broadcast style because of the limited transmission radius of base station.

3.2. Link Quality Model. In this section, the relation between link quality q and transmitting power P_t is introduced. Specifically, link quality is measured by packet reception rate (PRR).

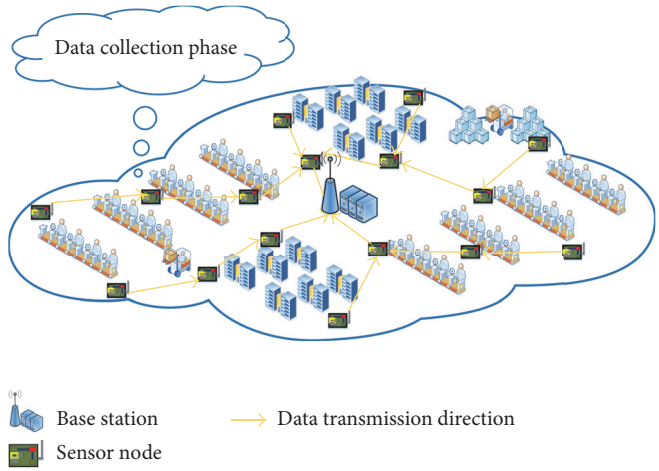


Figure 1: Industrial wireless sensor network.

In [30], Zuniga and Krishnamachari analyzed parameters on channel and proposed a mathematical formula to calculate packet reception rate (PRR):

$$q = \left(1 - \frac{1}{2}\exp^{-(\gamma/2)(B_N/R_D)}\right)^{8f}, \quad (1)$$

where R_D is data rate in bits, B_N is the noise bandwidth, and f is the frame size. SNR (signal-to-noise ratio) γ can be calculated through (2) if given specific transmitting power P_t and distance between transmitter and receiver d.

$$\gamma_{\mathrm{dB}}(d) = P_t - \mathrm{PL}(d_0) - 10n\log_{10}\left(\frac{d}{d_0}\right) - N(0,\sigma) - P_n, \quad (2)$$

where d_0 is a reference distance and n is the path loss exponent. Its value can be set to 2–4 if transmission approximately follows free space model. $\mathrm{PL}(d_0)$ and $10n\log_{10}(d/d_0)$ indicate attenuation caused by the adopted log-normal shadowing path loss model [31, 32]. In detail, path loss is influenced by signal diffusion and characteristics of channel while shadowing effect is caused by obstacles between transmitter and receiver [33]. $N(0,\sigma)$ is a zero-mean Gaussian RV with standard derivation σ and P_n is noise floor. Their value can be obtained through empirical measurements. Curves in Figure 2 show relation between packet reception rate and d under different transmitting power P_t in a static environment ($\sigma = 0$).

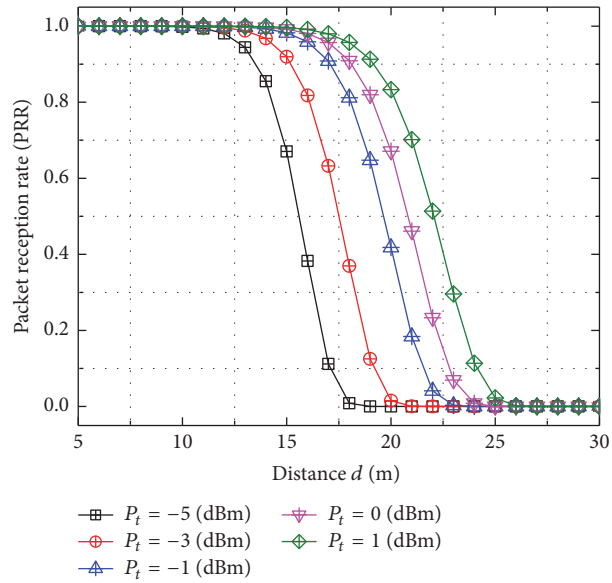


Figure 2: Relation between PRR and distance d (R_D = 19.2 kbps, B_N = 30 kHz, and n = 4).

3.3. Link Correlation Model. Link correlation model is used to obtain link correlation before codes dissemination phase. In this model, link correlation is obtained through broadcasting HELLO messages and reception vectors are used to keep information on receptions. One simple example shown in Figure 3 is used to demonstrate the construction of reception vectors and the calculation of link correlation.

First, nodes A and B broadcast 10 HELLO messages to one-hop downstream sensors in their broadcast domain (C, D, and E) separately. After successfully receiving one of these HELLO messages, C, D, and E will reply with an ACK (ACKnowledgement sent by receivers to confirm that data has been received successfully). Second, A and B construct reception vectors for C, D, and E according to these ACKs: if A receives the ACK for the ith HELLO message from C, then the ith element in the reception vector that A constructs for C will be 1. Otherwise, the ith element will be 0 because of the loss of HELLO message or ACK during transmission. Link reliability is not guaranteed in these two steps in order to reflect link correlation correctly (link reliability indicates that all packets can be received successfully through schemes like retransmission). Third, A and B will broadcast packets that contain information on these reception vectors to C,

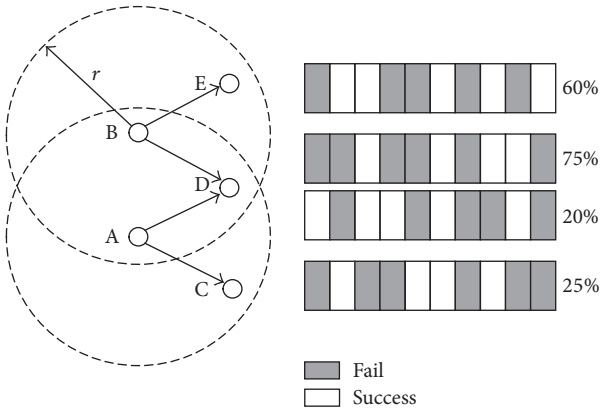


Figure 3: The illustration of link correlation.

D, and E. Upon receiving these packets, C, D, and E will use (3) to calculate link correlation C between themselves and A and B.

$$C(i,k) = \frac{\sum_{m=1}^{N} V_1(m) \times V_2(m) \times \cdots \times V_X(m)}{\sum_{m=1}^{N} V_k(m)}, \quad (3)$$

where N is the length of reception vector (also the number of HELLO messages), $V_k(m)$ is the mth element in the reception vector of node k, and i is the ID of node that broadcasts HELLO messages. In particular, $V_1(m) \times V_2(m) \times \cdots \times V_X(m)$ denotes the AND result of mth elements in reception vectors of all one-hop downstream nodes in the broadcast domain, which indicates that only when a HELLO message is received by all one-hop downstream nodes will the link correlation increase. For example, after A and B construct reception vectors for C and D and D and E separately and broadcast them, D will receive the reception vectors of C, D, and E as its location is covered by broadcast domains of A and B simultaneously. Then D uses (3) to calculate link correlations C(A, D) and C(B, D). Specifically, C(A, D) is $1/5 = 20\%$, while C(B, D) is $3/4 = 75\%$.

3.4. Energy Consumption Model. Two important sources of energy consumption are transmitting and receiving data. Their energy cost E_t and E_r can be estimated as follows:

$$E_t = P_t \times \frac{D^t}{R_D}, \qquad E_r = P_r \times \frac{D^r}{R_D}, \quad (4)$$

where P_t and P_r are transmitting power and receiving power separately. D^t and D^r denote data size that needs to be transmitted and received, and R_D is data rate.

3.5. Problem Statements. Delay-aware program dissemination (DAPD) focuses on reducing transmission delay during disseminating codes under the premise that the lifetime of network will not be influenced. Therefore, dissemination completion time (DCT) and residual energy in sensors are two main concerns. Problem statements are as follows.

(1) To Minimize the Dissemination Completion Time. Codes dissemination phase starts from the base station broadcasting code packets and ends until all active sensors in the network receive codes correctly. One aim of DAPD is to minimize time spent between these two time points.

(2) To Avoid Influencing the Lifetime of Network. Since one important operation in DAPD is to enhance transmitting power by utilizing excess energy, an overuse will definitely reduce the lifetime of network. Therefore, DAPD is designed based on two principles. (1) Available residual energy during codes dissemination phase in sensors is the difference between residual energy of themselves E_i and minimum residual energy in the network $E_{\min}$ before this phase starts. (2) Residual energy distribution should be as uniform as possible after codes dissemination phase. Two statements above can be expressed as follows:

$$\max\left(E_i - E_{\min}\right), \quad i = 1, 2, \ldots, M, \qquad \min\left(\sum_{i=2}^{M} \left(El_i - El_1\right)^2\right), \quad (5)$$

where E_i is the residual energy of sensor node i, $E_{\min}$ is minimum residual energy in the network before disseminating codes, M is the total number of active sensors, and El_i is the left energy of sensor node i after codes dissemination phase.

4. Scheme Design

4.1. Motivation

4.1.1. Unbalanced Energy Distribution. When the wireless sensor network is in the phase of data collection, neighbor sensors of the base station need to forward data originated in sensors far from the base station apart from transmitting back their own data, which leads to an unbalanced distribution of data load and energy consumption. Figure 4 shows the unbalanced distribution of residual energy in sensors with different distance from the base station after data collection phase. Furthermore, the lifetime of wireless sensor network can be defined as time that the network goes through until any sensor runs out of its energy (the first failure) [34]. Therefore, far nodes tend to have much underutilized energy during the lifetime of wireless sensor network if no additional scheme is taken.

Studies show that such residual energy can take up more than half of total energy in the network [34–36]. Hence, we consider taking advantage of residual energy in far nodes to enhance transmitting power during codes dissemination, and such enhancement will lead to a better link quality. Figure 5 presents the improvement on link quality after enhancing transmitting power, where link quality improves greatly when transmitting power is enhanced from −4 (dBm) to 4 (dBm); therefore the performance of codes dissemination can be improved.

With a more reliable link, number of retransmission and transmission delays can be reduced. Figure 6 is an illustration of expected dissemination completion time (DCT) for one

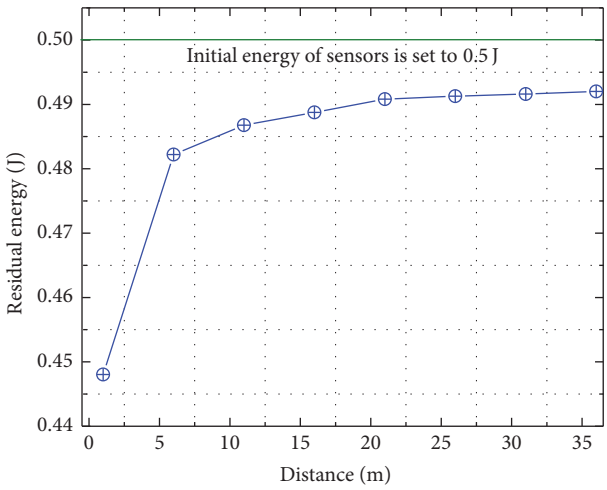


Figure 4: Distribution of residual energy on sensors.

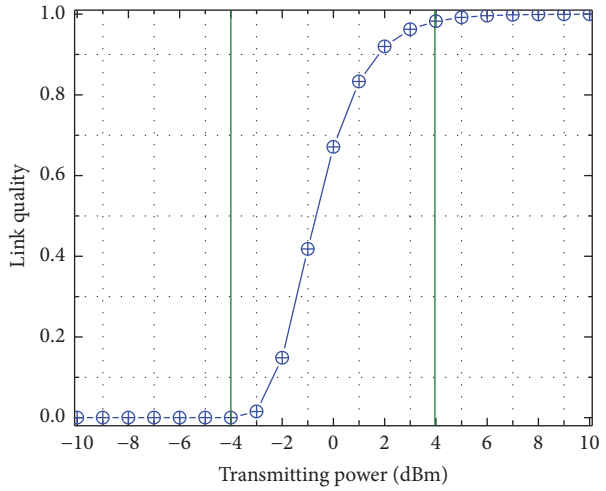


Figure 5: Relation between transmitting power and link quality (R_D = 19.2 kbps, B_N = 30 kHz, and n = 4).

sensor node to transmit code packets to its child nodes with different transmitting power.

Based on the analysis above, we can conclude that residual energy in sensors far from the base station can be used to enhance transmitting power during codes dissemination, and thus the performance of codes dissemination can be improved.

4.1.2. Change on Link Correlation. The utilization of link correlation during codes dissemination has already been proven to be fast and energy-efficient. However, experiments also show that the performance of correlated dissemination will degrade and be no better than deluge when in a highly dynamic environment. In detail, previous choices on parent nodes become outdated because link correlation will change over environment. The example in Figure 7 is used to illustrate the impact of changes on link correlation and the necessity on reselecting parent nodes.

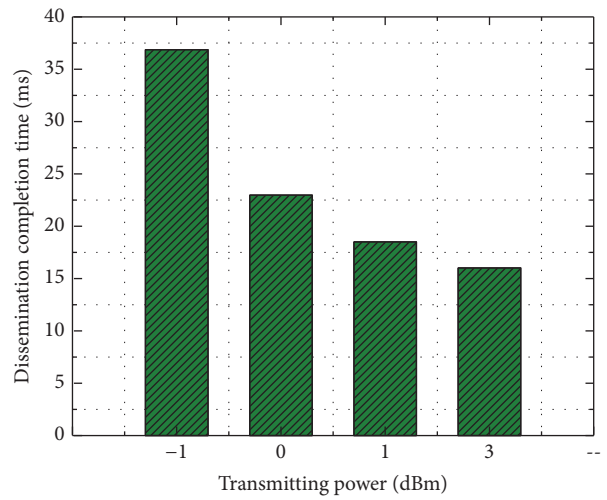


Figure 6: Expected dissemination completion time with different transmitting power.

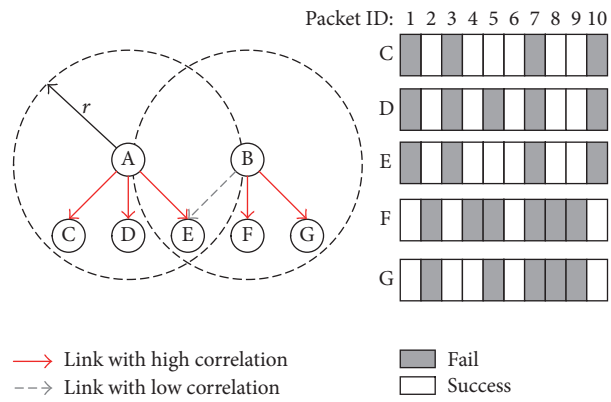


Figure 7: One example where A and B broadcast code packets to C, D, E, F, and G.

Initially, C, D, and E have already chosen A to be their parent node according to link correlation, while B has been chosen by F and G. After that, A and B start to broadcast code packets (10 packets one time) and will not broadcast following packets until these packets are correctly received by all their child nodes. Receptions on these packets are shown in the right part of Figure 7. The conclusion that link correlation between C, D, and E (or F and G) is high can be concluded from receptions. To simplify the illustration, we assume that all following packets retransmitted by A and B will be successfully received by C, D, E, F, and G. Therefore, A needs to retransmit 5 packets (ID: 1, 3, 5, 7, and 10) to C, D, and E, while B needs to retransmit 6 packets (ID: 2, 4, 5, 7, 8, and 9) to F and G.

Transmission on the next 10 packets will be similar to Figure 7 if the environment stands stable. However, this is not the case in the real world. One actual case after finishing transmitting the first 10 packets is shown in Figure 8.

The environment changes and link correlation C(B, E) is now higher than C(A, E), which can be seen from receptions on the following 10 code packets. If the parent node of C, D,

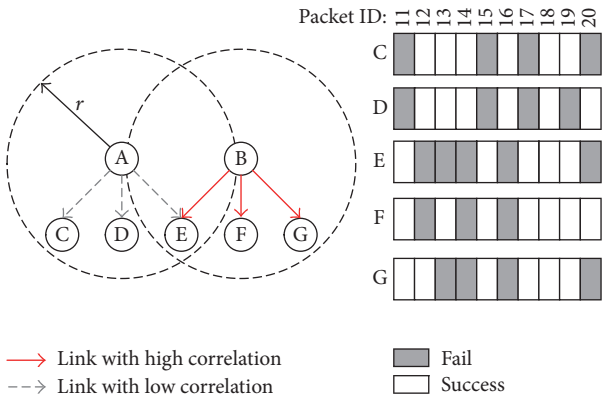


Figure 8: Link correlation changes after all child nodes correctly receive the first 10 code packets.

and E is still A, then A needs to retransmit 9 packets (ID: 11, 12, 13, 14, 15, 16, 17, 19, and 20). However, the performance will be improved if E reselects B as its parent node. Specifically, A will need to retransmit 5 packets (ID: 11, 15, 17, 19, and 20) to its child nodes (C and D), while B will need to retransmit 5 packets (ID: 12, 13, 14, 16, and 20) in this case. Hence, an operation that reselects parent nodes will lead to a better utilization of link correlation in the real world. Besides, compared with disseminating codes, cost on selecting parent node is relatively smaller. Therefore, it is possible to make better use of link correlation at the expense of going through another phase to reselect parent node when the environment is highly dynamic.

4.2. Design on Enhancing Transmitting Power. In order to enhance transmitting power without reducing the lifetime of wireless sensor network, we first need to correctly estimate residual energy in sensors before disseminating codes.

First, we analyze data load on each sensor during data collection phase. Timeout retransmission mechanism is adopted in our network model to ensure that all collected data can be sent back to base station correctly. Necessary notations are given as follows:

i, j, k: sensor node ID

h_i: hop count of node i

P_i^t: transmitting power of i

P_i^r: receiving power of i

q_{ij}: link quality between i (transmitter) and j (receiver)

N_i: number of packets that i needs to transmit to its one-hop upstream node

S_{DATA}: size of data packets

S_{ACK}: size of ACKs

Besides, as shown in Figure 9, i, j, and k are 3 neighbor sensors with different hop counts ($h_k = h_j + 1$; $h_j = h_i + 1$).

For example, j needs to receive N_k packets from k and transmit N_k ACKs back to k. Expected number of transmission is $1/q_{jk}q_{kj}$ because of the unreliability of link. On the other hand, j needs to transmit N_j packets to i and receive N_j ACKs from i and expected number of transmissions is $1/q_{ij}q_{ji}$. Therefore, we can obtain the amount of data that j needs to transmit D_j^t and receive D_j^r under the premise that N_k and N_j are known, which are shown in (6) and (7) separately. Besides, N_k and N_j can be calculated through (8) [37]. Equation (8) successfully estimates data load on sensor nodes with different distances from the base station in Send-Wait style with ACK protocol and no packet loss.

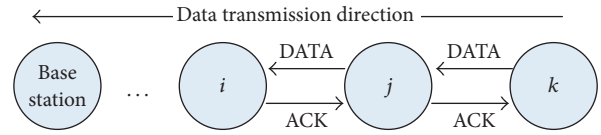


Figure 9: Three neighbor sensors, i, j, and k, with different hop counts during data collection phase.

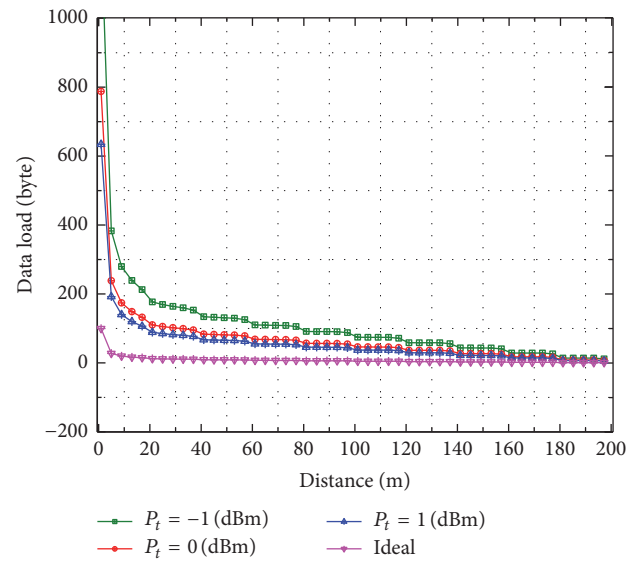


Figure 10: Data load on sensors with different distances from the base station.

$$D_j^t = \frac{N_k}{q_{jk}q_{kj}} \times S_{\text{ACK}} + \frac{N_j}{q_{ij}q_{ji}} \times S_{\text{DATA}}, \tag{6}$$

$$D_j^r = N_k \times S_{\text{DATA}} + N_j \times S_{\text{ACK}}, \tag{7}$$

$$N_i = (z+1) + \frac{z(z+1)r}{2l}, \tag{8}$$

where l is the distance between i and the base station, r is the transmission radius, and z is the largest integer that satisfies $zr + l < R$ (radius of the network). Figure 10 shows data load on sensors with $r = 20$ m in a wireless sensor network with $R = 200$ m.

Second, residual energy in sensors E can be estimated according to the data load above, combined with energy consumption model in Section 3.4.

$$E = E_0 - n(E_t + E_r), \tag{9}$$

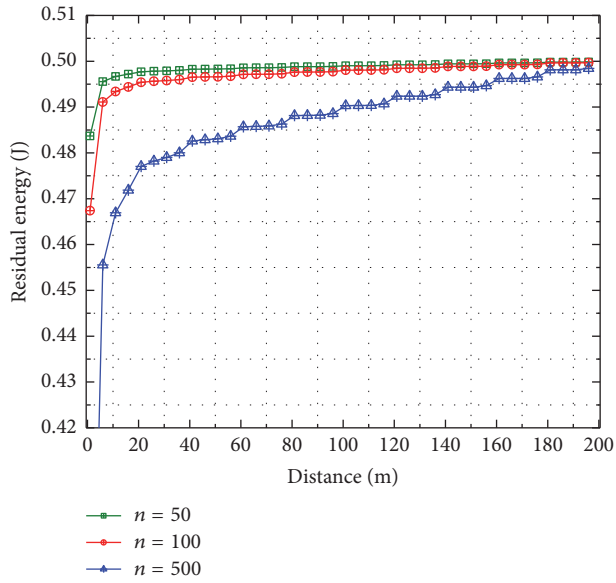


Figure 11: Residual energy in sensors after collecting data for different times.

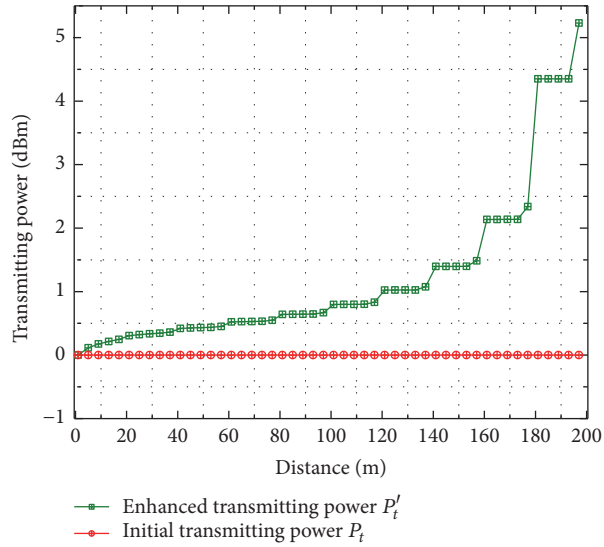


Figure 12: The enhancement of transmitting power.

where E_0 is initial energy in sensors and n is times of data collection. Figure 11 shows residual energy in sensors after collecting data for 50, 100, and 500 times.

To avoid the lifetime of network being influenced by the enhancement of transmitting power, an upper limit should be set according to sensors that have minimum energy left after data collection phase. From Figure 11, we can observe that such sensors tend to be deployed around the base station. Therefore, residual energy that can be used to enhance transmitting power E_a can be calculated through the following equation:

$$E_a = E_i - E_{\min}, \tag{10}$$

where E_i is the residual energy in sensor i and $E_{\min}$ is minimum residual energy in the network. E_a will be used to enhance transmitting power during the phase that disseminates codes and the phase that reselects parent nodes. Therefore, an additional variable α is introduced in order to allocate E_a to these two phases properly. For example, enhanced transmitting power P_t' of sensor i during codes dissemination is calculated according to the following equation:

$$P_t' = P_t + \frac{\alpha\left(E_i - E_{\min}\right)}{D} \times R_D \times \mathrm{PRR}_0\left(P_t\right), \tag{11}$$

where D is total data size on code packets, P_t is the initial transmitting power, and $\mathrm{PRR}_0(P_t)$ is the link quality under initial P_t. Due to the fact that the amount of data that needs to be transmitted in the phase that reselects parent node is smaller than that of codes dissemination phase, α should be larger than 0.5. Figure 12 shows the enhanced transmitting power of sensors in a case where D is 500 bytes and codes dissemination starts after collecting data for 50 times.

4.3. Our Methodology. In this section, we will show design details on delay-aware program dissemination (DAPD).

4.3.1. Overview. Delay-aware program dissemination (DAPD) is one kind of bulk data dissemination and it has 3 salient characteristics. (1) It exploits excess energy of sensors to enhance transmitting power during codes dissemination. (2) It exploits link correlation to reduce dissemination completion time. (3) It selectively goes through fast parent node reselection phase to quickly adapt to changes on environment and recover the improvement brought by link correlation. DAPD is composed of three phases: initial parent node selection phase, codes dissemination phase, and fast parent node reselection phase. Following sections will show detailed information on these three phases.

4.3.2. Initial Parent Node Selection Phase. This phase is used to obtain link correlation and choose parent node according to link correlation before disseminating codes.

First, the base station will initiate a flooding which enables each sensor node to obtain its hop count. Second, each node broadcasts HELLO messages to all one-hop downstream nodes in its broadcast domain, which contain its own ID and hop count. These one-hop downstream nodes will reply with ACKs upon successfully receiving HELLO messages. Link reliabilities are not guaranteed here in order to reflect link correlation correctly. Third, nodes which broadcast HELLO messages construct reception vectors for one-hop downstream nodes. Detailed information on the construction of reception vector is shown in Section 3.3. At last, these reception vectors will be broadcast to one-hop downstream nodes and upon receiving packets that contain information on reception vectors, these one-hop downstream nodes will calculate link correlation between themselves and the transmitter according to (3).

Due to the fact that the location of these one-hop downstream nodes may be covered by more than one transmitters' broadcast domain, they may receive reception vectors from many transmitters. They will choose the transmitter with the highest link correlation to be their parent node and

send a CHOSEN message to inform this node. To ensure link reliability, timeout retransmission mechanism is adopted during transmission on reception vectors and CHOSEN message.

For example, C will receive reception vectors of itself and D from A in Figure 3. Therefore, C only needs to calculate link correlation for one time. (C(A, C) = 1/4 = 25%) and it has to choose A as its parent node regardless of link correlation because C is only covered by the broadcast domain of A. However, D will receive reception vectors of all one-hop downstream nodes (C, D, and E). After using (3) to calculate link correlations between itself and A and B, it chooses B to be its parent node because C(B, D) is much larger than C(A, D).

4.3.3. Codes Dissemination Phase. After initial parent node selection phase, each sensor node in the network will obtain its parent node ID and all its child nodes ID. Next, codes dissemination phase is initiated by the base station broadcasting code packets. Sensor nodes also start broadcasting code packets to child nodes after successfully receiving all packets. During broadcasting, nodes will continuously broadcast N packets at a time and will not broadcast following packets until all child nodes' ACKs for these packets are received.

Besides, one node may receive packets from other one-hop upstream nodes apart from its parent node. In this case, it will compare the packet ID with N packet IDs that it can currently receive from its parent node. (1) The packet is not one of those N packets or has already been successfully received; then it will discard this packet. (2) The packet is one of those N packets and has not been received; then it will receive this packet and reply an ACK for this packet to its own parent node. Such mechanism will reduce dissemination completion time further. Code packets are transmitted according to operations above hop by hop until all sensor nodes receive codes successfully.

4.3.4. Fast Parent Node Reselection Phase. Section 4.1.2 shows that changes on environment during codes dissemination phase will lead to degradation on the performance of correlated dissemination. Current link correlation may be very different from the link correlation calculated before disseminating. Therefore, previous choices on parent node can be outdated. To reuse link correlation, another parent node selection phase is necessary. However, unlike Section 4.3.2 which initiates from the base station, a parent node selection phase that only happens between nodes and their one-hop downstream nodes is needed here. In detail, one sensor node will rebroadcast HELLO messages to all one-hop downstream nodes in its broadcast domain before transmitting code packets. Then, again, these one-hop downstream nodes will reply ACKs for each HELLO message. The following steps are same to initial parent node selection phase. Figure 13 shows the data transmission for one-hop downstream nodes (i + 1) to choose parent nodes (i), where V indicates packets that contain information on reception vectors; τ is extra time to make sure all potential ACKs can arrive.

Since the incentive for going through this reselection phase is to reuse link correlation, therefore, it is necessary for

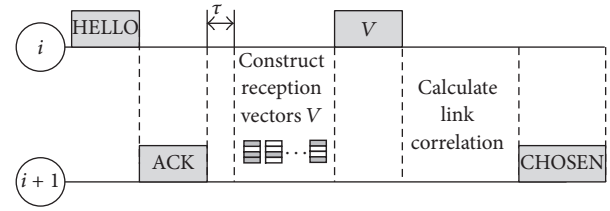


Figure 13: Data transmission during fast parent node reselection phase.

sensor nodes to keep monitoring on link correlation during codes dissemination phase. When link correlation drops below a predefined threshold, one sensor node can make the assumption that environment around itself has changed and a fast parent node reselection phase is needed before transmitting code packets to child nodes. However, it can be time-consuming and unreasonable to obtain link correlation through the same way described in Section 4.3.2 (needs parent node's participation). The method we adopted here to make quick estimations on link correlation is collecting statistics on the percent of unneeded retransmitted code packets. In detail, after finishing one round of transmission or retransmission, (12) is used to estimate link correlation:

$$s = 1 - \frac{k_u}{k_{\text{all}}}, \tag{12}$$

where k_u is the number of unneeded retransmitted code packets, k_{all} is the total number of received code packets, and s indicates the percent of useful retransmitted code packets for one sensor node. When s is lower than a predefined threshold, one sensor node will go through a fast parent node reselection phase before disseminating code packets to its child nodes. Take Figures 7 and 8 as an example: after the first round of retransmission, node E will use (12) to estimate link correlation. The value of s in Figure 7 is (1 − 1/5) = 80%, while that in Figure 8 is (1 − 4/9) = 56%. Therefore, it is more likely for node E in Figure 8 to go through a reselection phase than node E in Figure 7.

Obviously, this phase will only prolong the dissemination completion time if link correlation remains stable during codes dissemination. However, this phase will improve the performance when the environment is highly dynamic. Therefore, it is necessary to achieve a balance between the improvement brought by this phase and extra delay results from this phase. Detailed analysis on delay is shown in Section 5.2 and its impact in our experiment is shown in Section 6.4.1.

4.3.5. The Delay-Aware Program Dissemination Algorithm. See Scheme 1.

5. Analysis on Delay

5.1. Analysis on Delay of Codes Dissemination. First, we analyze transmission delay for one node to broadcast N code packets to its child nodes. Changes on link correlation

Initialize: Each sensor node obtains its hop count through a flooding originated from the base station;
Initial parent node selection phase
(1) **For** each node i **Do**
(2) Broadcast HELLO messages;
(3) **For** each node j who receives HELLO messages above **Do**
(4) Reply ACKs to i;
(5) **End for**
(6) Construct reception vectors for each node j;
(7) Broadcast reception vectors;
(8) **For** each node k who has the highest correlation with node i
(9) Send a CHOSEN message to node i;
(10) **End for**
(11) **End for**
Data dissemination phase
(1) **For** $h = 0$ to $X - 1$ (h: hop count; X: the largest hop count in the network)
(2) Use equation (12) to estimate link correlation;
(3) **If** $s >$ Threshold
(4) **For** each node i whose hop count is h **Do**
(5) Broadcast code packets;
(6) **For** each node j who receives packets above and its hop count equals $h + 1$ **Do**
(7) **If** the parent node of j is i
(8) Reply ACKs to i;
(9) **Else**
(10) Reply ACKs to its own parent;
(11) **End for**
(12) **End for**
(13) **Else**
(14) Go through the next phase and then return Step (4);
(15) **End for**
Fast parent node reselection phase
(1) **For** each node i whose hop count is h **Do**
(2) Broadcast HELLO messages;
(3) **For** each node j who receives HELLO messages above **Do**
(4) **if** hop count of j is $h + 1$
(5) Reply ACKs to i;
(6) **End for**
(7–11) Same to the first phase;

Scheme 1: Delay-aware program dissemination scheme.

during codes dissemination are ignored here and necessary notations are given as follows:

i: parent node

X: number of child nodes

n_k: child node ID ($k = 1, 2, \ldots, X$)

q_i: link quality when i transmits packets

q_{n_k}: link quality when k replies ACKs

N: number of code packets that i transmits

c_{n_k}: link correlation between n_k and i

According to (3), we can obtain number of code packets that are successfully received by all child nodes at the first transmission.

$$\sum_{m=1}^{N} V_{n_k}(m) = Nq_{n_k}, \tag{13}$$

$$\sum_{m=1}^{N} \left(V_{n_1}(m) \times V_{n_2}(m) \times \cdots \times V_{n_X}(m)\right) = Nc_{n_k}q_{n_k}, \tag{14}$$

where $V_{n_k}(m)$ is the reception on the mth packet of child node n_k. The left part of (13) indicates number of packets that need not to be retransmitted.

According to (14), we also have

$$c_{n_1}q_{n_1} = c_{n_2}q_{n_2} = \cdots = c_{n_X}q_{n_X}. \tag{15}$$

Therefore, the value of k is irrelevant to number of packets that are successfully received by all child nodes in (13), which is a variable shared by all child nodes.

At the first time, i needs to transmit all N code packets.

At the second time, the number is

$$N - Nc_{n_k}q_{n_k}. \tag{16}$$

And, at the third time, the number is

$$\begin{aligned} & N - Nc_{n_k}q_{n_k} - \left(N - Nc_{n_k}q_{n_k}\right) \times c_{n_k}q_{n_k} \\ & \quad = N - 2Nc_{n_k}q_{n_k} + N\left(c_{n_k}q_{n_k}\right)^2 \\ & \quad = N - 2Nc_{n_k}q_{n_k} + o\left(c_{n_k}q_{n_k}\right) \approx N - 2Nc_{n_k}q_{n_k}. \end{aligned} \tag{17}$$

Besides, the expected number of transmissions T is $1/\min(q_i q_{n_k})$. Hence, total number of code packets that parent node i needs to transmit can be obtained through the following equation:

$$D = \sum_{m=1}^{T} \left(N - N\left(m-1\right)c_{n_k}q_{n_k}\right). \tag{18}$$

However, calculated number of packets that need to be transmitted above can be imprecise in the real world, since link correlation will change during codes dissemination and link quality will not always be the same to the link quality estimated by (1).

After obtaining D, transmission delay can also be calculated. A complete process of transmission is described as follows. First, parent node i broadcasts N code packets continuously and starts to receive ACKs from its child nodes. Upon receiving some or all of these packets, one child node will reply with ACKs that contain information on which packets are successfully received. After keeping receiving ACKs for a long time which ensures that all potential ACKs can arrive at the parent node, i will broadcast left packets which are not successfully received by all child nodes. The process described above will cycle until all its child nodes receive N code packets; then parent node i will start to broadcast the next N packets. Figure 14 is a sequence diagram of transmission.

t_{DATA} is time spent on transmitting code packets, t_{ACK} is time spent on replying ACKs, and τ is an extra time to ensure that all potential ACKs can arrive. Hence, the delay for one node to broadcast N code packets to its child nodes is

$$t = \sum_{m=1}^{T} \left(\frac{D_m \times S_{\mathrm{DATA}}}{R_D} + \frac{D_m \times S_{\mathrm{ACK}}}{R_D} + \tau\right), \tag{19}$$

where D_m is number of code packets that need to be transmitted at the mth time, S_{DATA} and S_{ACK} are size of code packet and ACK separately, and R_D is data transmission rate. Figure 15 shows transmission delay for one node to transmit ten code packets to all its one-hop downstream nodes in several cases where link correlation between these nodes is set to 20%, 40%, 60%, and 80% manually. Transmitting power of nodes in Figure 15 is same to the green curve in Figure 12, while S_{DATA}, S_{ACK}, and R_D are same to Section 6.1. With the distance from the base station increasing, gaps between transmission delay start to shrink, since a more reliable link tend to weaken benefits brought by link correlation.

5.2. Analysis on Delay of Parent Node Reselection. Due to the fact that the environment of wireless sensor network

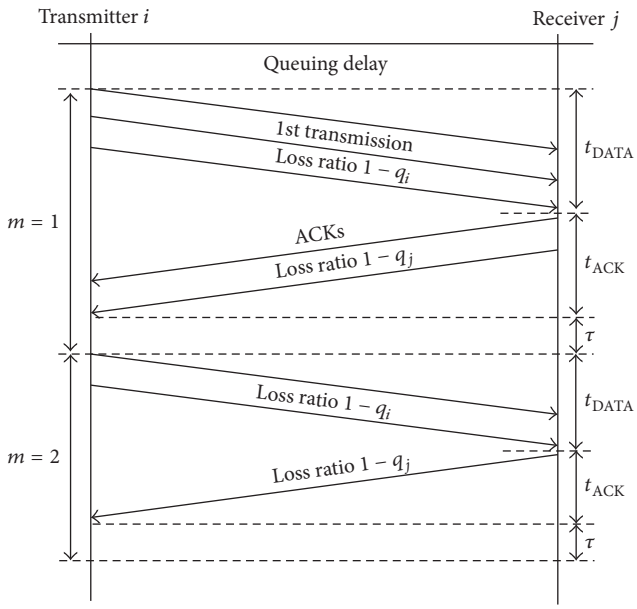


Figure 14: Sequence diagram of transmission.

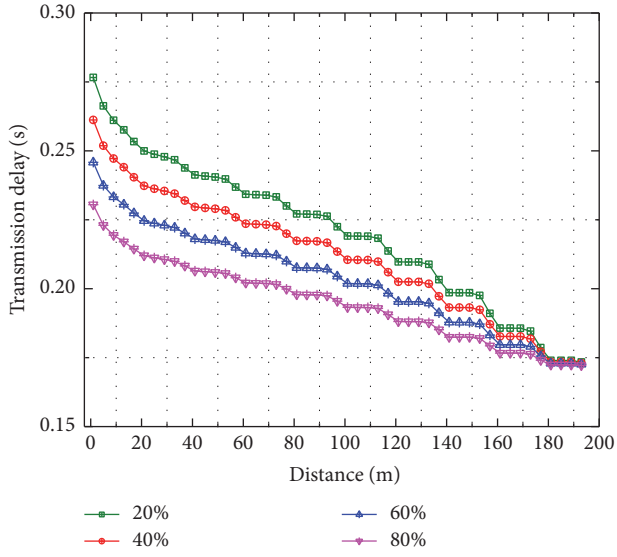


Figure 15: Transmission delay with different link correlation.

can be highly dynamic, link correlation obtained in initial parent node selection phase tends to change during codes dissemination phase. An example in Section 4.1.2 shows that the improvement brought by adopting link correlation can be greater after reselecting parent nodes during codes dissemination. However, this new parent node selection phase will also bring extra delay to codes dissemination. Therefore, the delay brought by fast parent node reselection phase is analyzed in this section.

Data transmission of this phase is shown in Figure 13. In this phase, reliabilities of HELLO messages and ACKs are not guaranteed, while that of V (packets contain information on reception vectors) and CHOSEN message are assured through timeout retransmission mechanism.

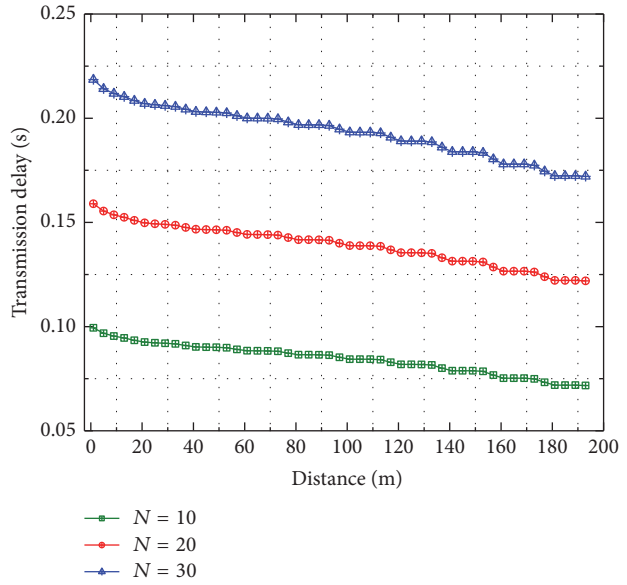


Figure 16: Expected delay brought by fast parent node selection phase.

First, sensor nodes continuously broadcast N HELLO messages to one-hop downstream sensors in their broadcast domain. Second, they start to receive ACKs sent by one-hop downstream nodes. An extra time τ is also added here to ensure the arrival of ACKs. Then, they construct reception vectors according to these ACKs and broadcast V, which is composed of packets that contain information on reception vectors. After receiving V, one-hop downstream nodes will calculate link correlations for each node that broadcasts V and select the node with which they have highest link correlation as theirs parent node. At last, a CHOSEN message is sent to inform this node that it has been chosen.

According to the description above, the expected delay brought by fast parent node reselection phase is shown in the following equation:

$$t_d = \frac{D_{\text{HELLO}} + D_{\text{ACK}}}{R_D} + \tau + T \times \left(\frac{D_V + D_{\text{CHOSEN}} + 2D_{\text{ACK}}}{R_D} + 2\tau \right), \quad (20)$$

where D_* is data size on corresponding packet and T is the expected number of transmissions. Figure 16 shows transmission delay for nodes to go through fast parent node reselection phase with N HELLO messages; additional delay brought by the phase reduces with distance from the base station increasing since a higher transmitting power also benefits this phase.

6. Experimental Evaluation

In this section, a simulation experiment is given to evaluate delay-aware program dissemination (DAPD). First, parameters of network and sensor nodes are introduced. Second, we conduct the experiment according to parameters above and experiment results are compared with deluge and Link-Correlation-Aware Data Dissemination (CD). Third, we analyze the impact of changes on network parameters.

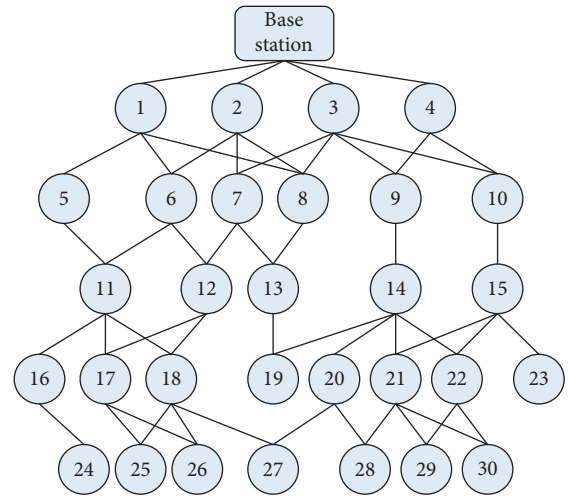


Figure 17: The topology of our network.

Table 1: Parameter setting on sensors.

Parameter	Value
Working voltage	2.7–3.3 (V)
Transmitting power	−20–10 (dBm)
Receive sensitivity	−101 (dBm)
Transmission rate	<76.8 (kbps)

6.1. Parameters Setting. The wireless sensor network is composed of one base station and thirty sensor nodes. Each node can have 1 to 4 links (represented by black line) with one-hop downstream nodes and the largest hop count in the network is set to 5. The topology of network is generated randomly with parameters above and final result is shown in Figure 17.

The energy of base station is considered to be infinite; therefore its transmitting power is high enough. Initial energy of sensors in the network is 0.5 J and transmission radius of all sensors is 20 m. Total data size on codes is 40 kB, while data sizes on each code packet S_{DATA} and ACK S_{ACK} are 20 bytes and 5 bytes separately. Besides, τ is designated as 5 (ms). Parameter settings on sensors are shown in Table 1.

To ensure that all sensor nodes can be updated successfully through codes, each sensor will broadcast 16 packets continuously and start to receive ACKs after then. Following packets will not be transmitted until it receives all its child nodes' ACKs for these 16 packets. Besides, change on link correlation during codes dissemination is ignored here and analyzed independently in Section 6.4.1.

The transmitting power of deluge and CD is shown in Table 2 and that of DAPD after enhancing without reducing the lifetime of network is shown in Table 3. Before reprogramming, they all have already collected data packets for 100 times and transmitted them back to the base station with $P_t = 0$ (dBm).

Table 2: Transmitting power of deluge and CD.

h	1	2	3	4	5
P_t (dBm)	0	0	0	0	0
q	0.6710	0.6710	0.6710	0.6710	0.6710

Table 3: Transmitting power of DAPD.

h	1	2	3	4	5
P_t (dBm)	0	0.1	0.8	1.8	3
q	0.6710	0.7109	0.8078	0.9069	0.9625

h is hop count, P_t is transmitting power, and q is theoretical link quality calculated through (1).

6.2. Selected Parent Nodes. First, each sensor node will select its parent node during initial parent node selection phase described in Section 4.3.2.

For example, reception vectors kept by nodes with hop count 1 are shown in Table 4.

P is the ID of transmitter, N is the ID of receiver, and $V(i)$ is the ith element in the reception vector (1 indicates a successfully received HELLO message, while 0 indicates a failure). Figure 18 shows selected parent nodes after this phase. Top of red lines denotes selected parents of nodes. Grey lines indicate that although those one-hop upstream sensors are not parent nodes, code packets broadcast by them are still possible to be received by nodes connected by the bottom of grey lines.

6.3. Comparison with Deluge and CD

6.3.1. Evaluation on Delay. Average transmission delay of deluge, CD, and DAPD is shown in Figure 19. i in x-axis indicates codes dissemination from sensors with hop count $i-1$ to sensors with hop count i. According to Figure 19, average transmission delay of DAPD is 19.05% and 16.65% smaller than deluge and CD separately. Two reasons can account for this improvement: (1) DAPD adopts link correlation compared with deluge; (2) DAPD intelligently enhances transmitting power to reduce delay further compared with CD. Figure 20 presents the distribution of link correlation and link quality collected from all reception vectors, from which we can observe that although link correlation does not present any regular distribution, link quality in DAPD is higher than CD appreciably due to the enhancement of transmitting power.

6.3.2. Evaluation on Energy Consumption. Two main sources of energy consumption come from transmitting and receiving code packets, ACKs. Therefore, we make comparisons on the sum of these two sources. Results are shown in Figure 21. Due to the fact that there is no need for sensors with hop count 5 to transmit code packets, their energy consumption is far smaller than upstream nodes and is therefore ignored. From Figure 21, we can see that on one hand energy consumption of DAPD will become larger with hop count increasing, which meets our scheme that utilizes excess energy in far nodes to enhance transmitting power; on the other hand, the energy consumption of both CD and DAPD is smaller than deluge on sensors with hop count smaller than 4, where link correlation plays a more important role. Therefore, the adoption of link correlation can reduce delay and energy consumption simultaneously. Besides, Figure 22 shows the distribution of residual energy after disseminating codes for one and three times; DAPD slightly relieves the unbalanced residual energy distribution of sensors.

6.4. Impact of Network Parameters

6.4.1. The Volatility of Environment. When the environment of wireless sensor network is highly dynamic, parent nodes selected during initial parent node selection phase may be outdated and unable to make full use of link correlation during codes dissemination phase. For example, Figure 23 shows a possible situation where average link correlations between sensors with hop count 3 and 4 and sensors with hop count 4 and 5 are modified from 62.47% and 74.63% to 20% manually. Average transmission delay of CD and DAPD increases toward deluge after changing link correlations manually.

Through using (12) to estimate link correlation, sensor nodes with hop count 3 and 4 will make the assumption that it is necessary to go through a fast parent node reselection phase. Data sizes on HELLO message, ACK, and CHOSEN message are all 5 bytes, while that on packet of V is same as S_{DATA} for convenience. Compared with time used for transmission, times spent on constructing reception vectors and calculating link correlation are far smaller; therefore they are ignored when measuring delay. Figure 24 shows the delay before and after adopting fast parent node reselection phase, from which we can see that although transmission delay cannot recover to previous unchanged level, it improves a lot compared with taking no action when link correlation changes. Besides, transmitting power of sensors becomes higher with distance from base station increasing, which leads to a lower delay, and this improvement also benefits the fast parent node reselection phase. Therefore, the delay brought by this phase will decrease when it happens on sensors with a large hop count.

6.4.2. Data Size and Packet Size. Figure 25 shows dissemination completion time (DCT) of codes dissemination with different packet size, while Figure 26 shows dissemination completion time (DCT) with different total size of data.

6.4.3. Number of Hop Counts and Nodes. The scale of wireless sensor network mainly depends on the number of hop counts and nodes, which tend to have great impacts on the delay. On one hand, the increase on hop count will directly increase number of times that sensor nodes transmit and retransmit packets. On the other hand, amount of data that

Table 4: Reception vectors kept by nodes.

P	N	$V(1)$	$V(2)$	$V(3)$	$V(4)$	$V(5)$	$V(6)$	$V(7)$	$V(8)$	$V(9)$	$V(10)$
①	⑤	1	1	0	1	1	0	0	1	1	1
①	⑥	0	1	1	1	1	0	1	1	1	1
①	⑧	1	0	1	1	1	1	1	1	1	0
②	⑥	1	0	0	0	0	1	1	0	1	0
②	⑦	1	1	1	1	0	1	1	1	1	1
②	⑧	0	1	1	0	0	1	1	0	1	0
③	⑦	1	0	1	1	0	0	1	0	0	0
③	⑧	1	0	1	0	1	1	0	0	1	1
③	⑨	1	1	1	1	1	0	1	1	1	1
③	⑩	0	0	1	1	1	0	1	1	0	1
④	⑨	0	1	0	1	0	1	0	1	1	1
④	⑩	1	0	0	0	0	1	1	1	0	1

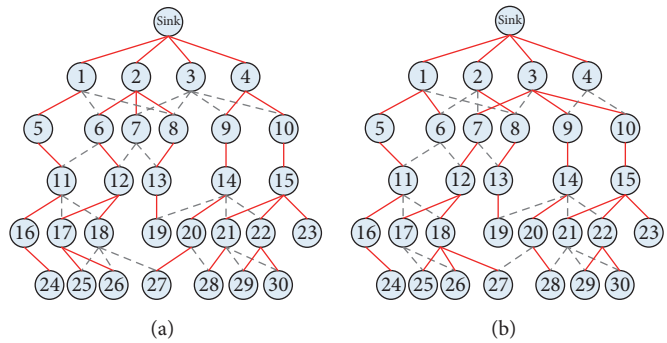


Figure 18: Selected parent nodes in CD and DAPD. (CD: (a); DAPD: (b)).

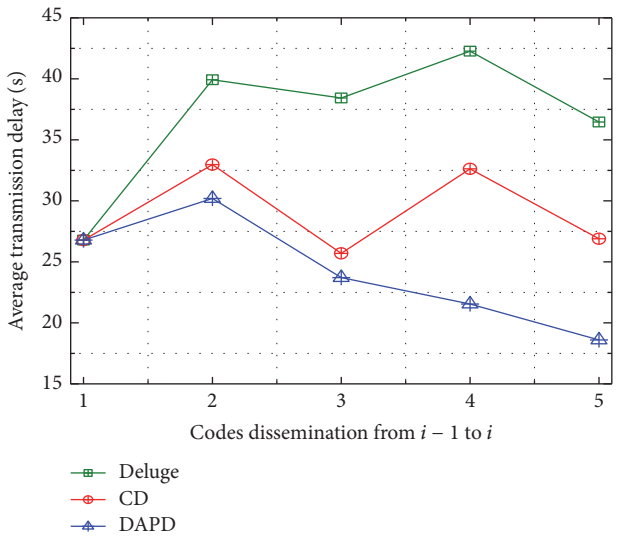


Figure 19: Average transmission delay.

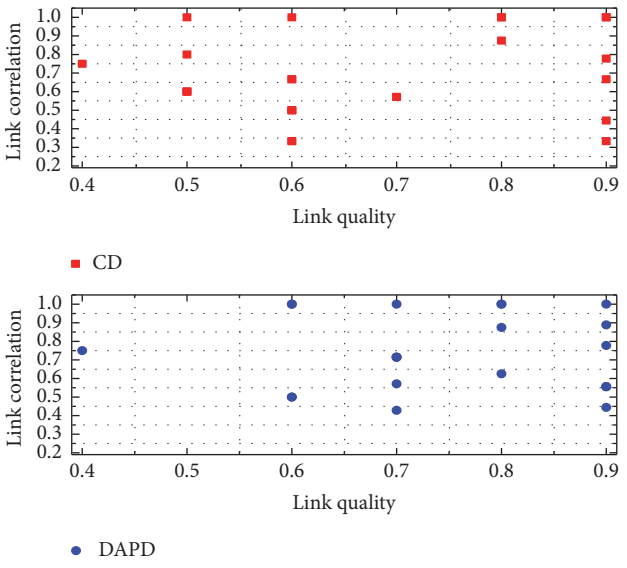


Figure 20: Distribution of link quality and link correlation.

generates during data collection phase will also increase with number of nodes becoming larger, which leads to more energy consumption and a more unbalanced residual energy distribution. Therefore, it is meaningful to analyze the impact of network's scale on our scheme. Since the distribution on link correlation is irregular according to Figure 20, the impact

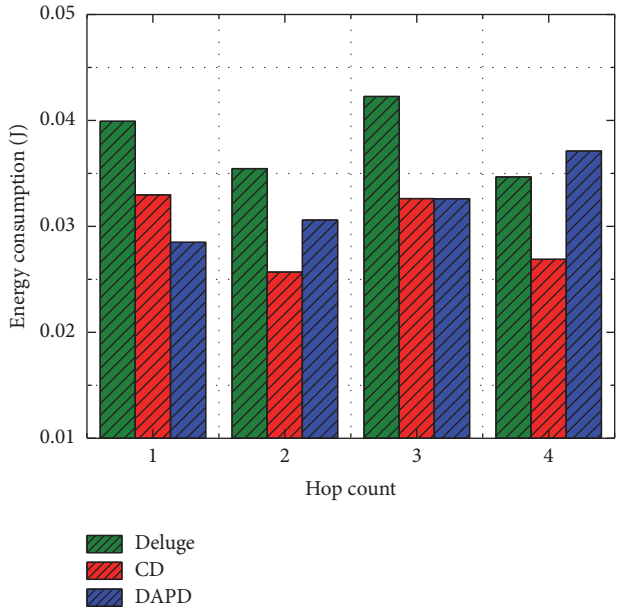


Figure 21: Energy consumption of deluge, CD, and DAPD.

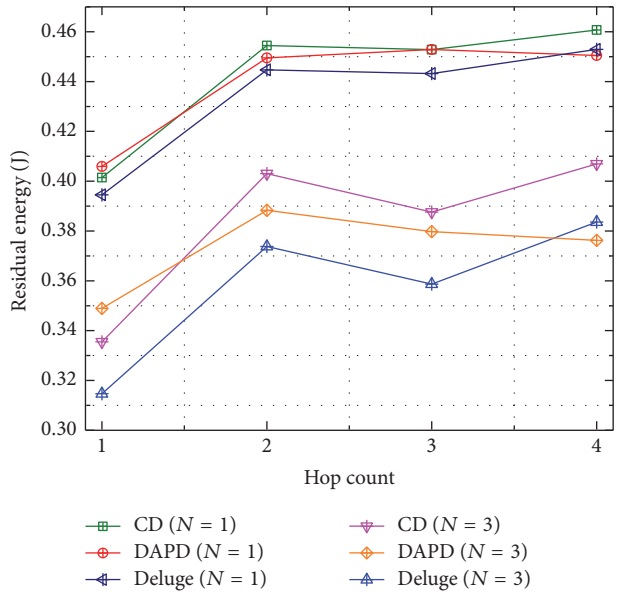


Figure 22: Distribution of residual energy.

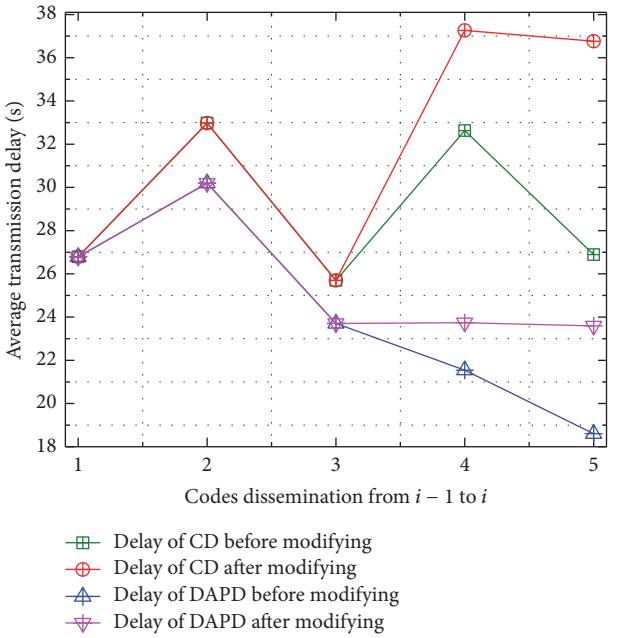


Figure 23: Delay of CD and DAPD after modifying link correlation manually.

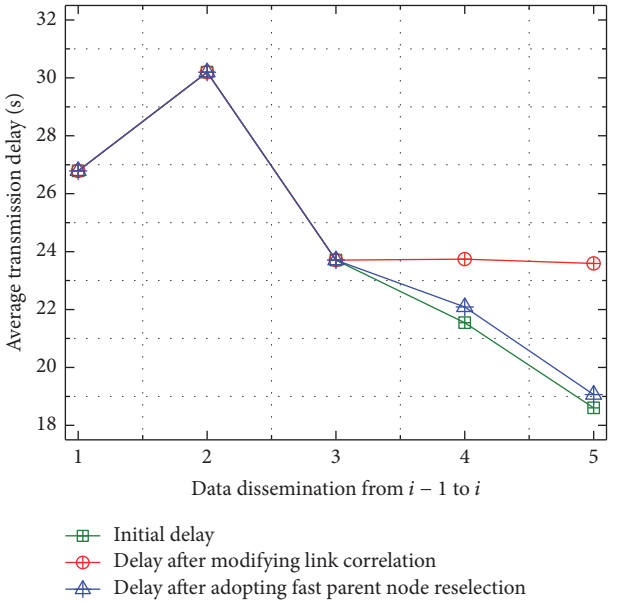


Figure 24: Delay for DAPD after adopting fast parent node reselection phase.

concentrates on the enhancement of transmitting power. First, Figure 27 shows data load on nodes with different distance from base station during data collection phase (ρ_0 is the original density of nodes); gaps between nodes around and nodes far from the base station become wider with number of hop count and nodes increasing.

With a different distribution of data load after the scale of network changing, transmitting power during codes dissemination phase will also be different from Table 3, which is shown in Figure 28.

6.4.4. The Period of Codes Dissemination. Apart from hop count and number of nodes, frequency of codes dissemination will also influence our scheme. Generally, the enhancement of transmitting power will be smaller with a higher frequency. The period T is defined as follows:

$$T = \frac{n_c}{n_d}, \tag{21}$$

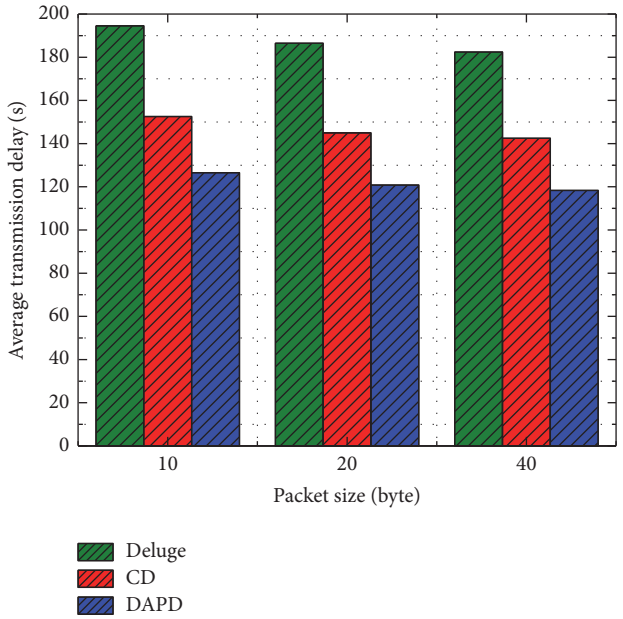


Figure 25: DCT with different packet size.

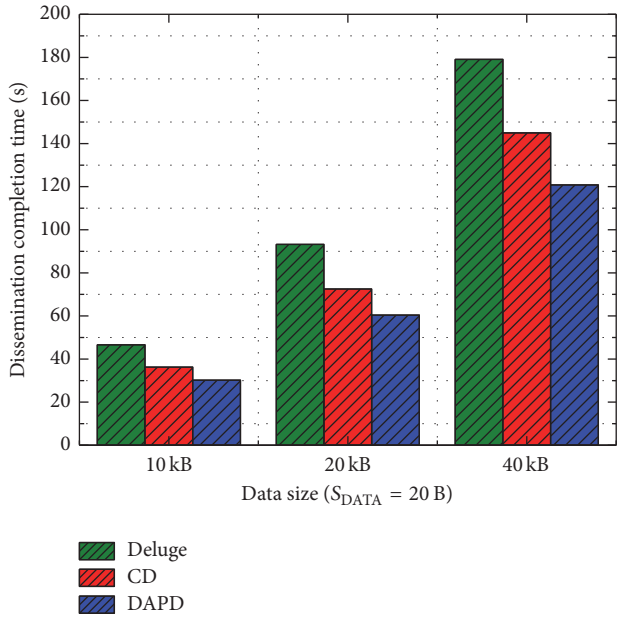


Figure 26: DCT with different total size of data.

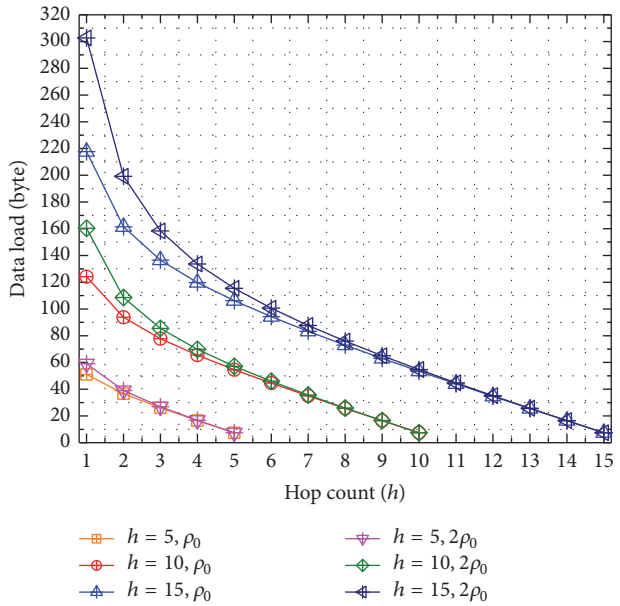


Figure 27: Data load on sensors in wireless sensor networks with different scales.

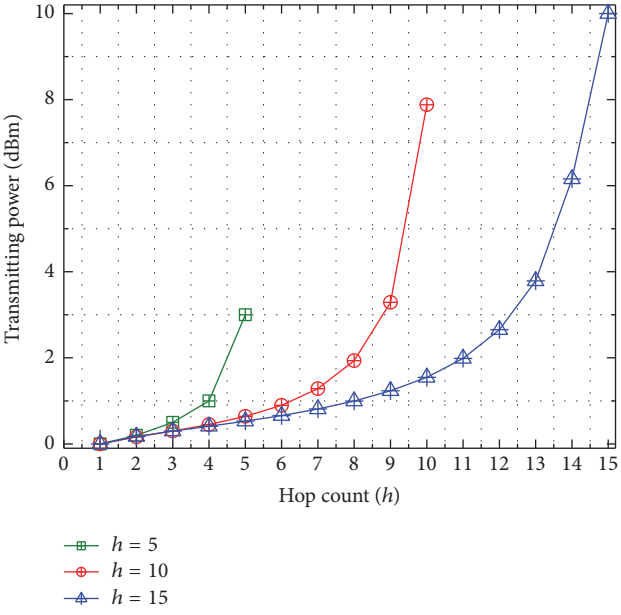


Figure 28: The enhancement of transmitting power in networks with different scales.

where n_c is times of collecting data instead of times of sending back data packets because the latter one is influenced by link quality and can be unpersuasive when measuring period and n_d is times of codes dissemination. Impacts of different periods on the enhancement of transmitting power are shown in Figure 29.

7. Conclusion

In this paper, we propose a novel codes dissemination scheme which focuses on reducing delay and being adaptive to highly dynamic environment. In general, it has three salient features: (1) utilize link correlation in wireless sensor network to reduce number of transmissions, (2) enhance transmitting power during codes dissemination to improve link quality, and (3) go through a fast parent node reselection phase when the environment changes in order to reuse link correlation. A simulation experiment shows that, compared with deluge and another codes dissemination scheme CD, our scheme achieves better performance in both static and dynamic environments.

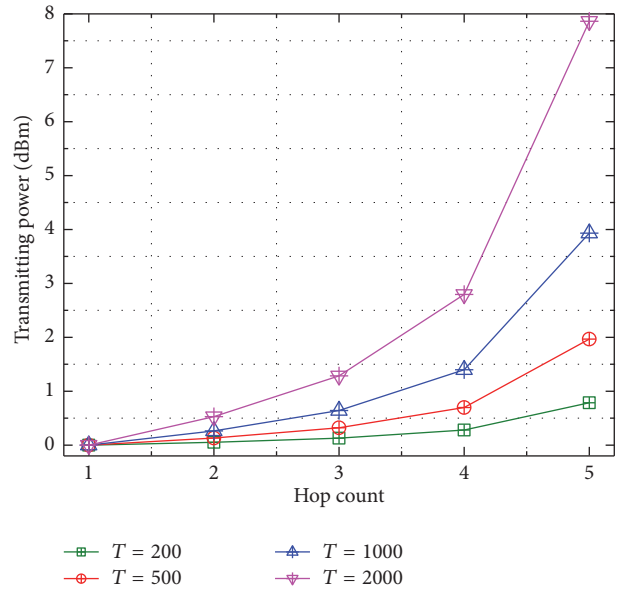


FIGURE 29: The enhancement of transmitting power with different period T.

However, as mentioned in Section 6.3.2, one disadvantage on DAPD is that residual energy in sensor nodes with largest hop count is not exploited, since there is no need for these nodes to broadcast code packets during codes dissemination phase, which is an important source of energy consumption for other nodes in the network.

Our future work includes integrating code packets transmission between sensor nodes with same hop count into DAPD to improve the performance further and a comprehensive research on the relation between link quality and link correlation.

Competing Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61379110, 61370229, and 61305144), the National Basic Research Program of China (973 Program) (2014CB046305), the National Key Technology R&D Program of China (2014BAH28F02), and the S&T Projects of Guangdong Province (2014B010103004, 2014B010117007, 2015A030401087, 2015B010110002, 2015B010109003, and 2016B010109008).

References

[1] M. Aazam and E.-N. Huh, "Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT," in *Proceedings of the IEEE 29th International Conference on Advanced Information Networking and Applications (AINA '15)*, pp. 687–694, IEEE, Gwangiu, South Korea, March 2015.

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[3] CISCO's Technology News Site, https://newsroom.cisco.com/ioe.

[4] M. Stauffer, "Connecting the internet of everything," in *Proceedings of the Hot Chips 26 Symposium (HCS '14)*, vol. 26, pp. 38–10, IEEE, Cupertino, Calif, USA, August 2014.

[5] S. He, J. Chen, X. Li et al., "Mobility and intruder prior information improving the barrier coverage of sparse sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1268–1282, 2014.

[6] H. Li, D. Liu, Y. Dai, and T. H. Luan, "Engineering searchable encryption of mobile cloud networks: when QoE meets QoP," *IEEE Wireless Communications*, vol. 22, no. 4, pp. 74–80, 2015.

[7] B. Fan-Yu, X. Wang, and Q. Zhang, *Data Ming Model Based on Cloud Computing in the Internet of Things*, Computer & Information Technology, 2012.

[8] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 312–325, 2016.

[9] K. Suto, H. Nishiyama, N. Kato, and CW. Huang, "An energy-efficient and delay-aware wireless computing system for industrial wireless sensor networks," *IEEE Access*, vol. 3, pp. 1026–1035, 2015.

[10] Z. Tang, A. Liu, and C. Huang, "Social-aware data collection scheme through opportunistic communication in vehicular mobile networks," *Included in Special Section in IEEE Access: Recent Advances in Socially-aware Mobile Networking*, vol. 4, pp. 6480–6502, 2016.

[11] X. Liu, T. Wei, and A. Liu, "Fast program codes dissemination for smart wireless software defined networks," *Scientific Programming*, vol. 2016, Article ID 6907231, 21 pages, 2016.

[12] 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.

[13] V. Sharma, I. You, and R. Kumar, "Energy efficient data dissemination in Multi-UAV coordinated wireless sensor networks," *Mobile Information Systems*, vol. 2016, Article ID 8475820, 13 pages, 2016.

[14] W. Dong, C. Chen, X. Liu, G. Teng, J. Bu, and Y. Liu, "Bulk data dissemination in wireless sensor networks: modeling and analysis," *Computer Networks*, vol. 56, no. 11, pp. 2664–2676, 2012.

[15] R. Adline Freeda and R. N. Sharmila, "A review of bulk data dissemination protocols for reprogramming in WSN," in *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES '16)*, pp. 1–4, IEEE, Chennai, India, Feburary 2016.

[16] Q. Wang, Y. Zhu, and L. Cheng, "Reprogramming wireless sensor networks: challenges and approaches," *IEEE Network*, vol. 20, no. 3, pp. 48–55, 2006.

[17] J. W. Hui and D. Culler, "The dynamic behavior of a data dissemination protocol for network programming at scale," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 81–94, November 2004.

[18] H. Lim and C. Kim, "Flooding in wireless ad hoc networks," *Computer Communications*, vol. 24, no. 3-4, pp. 353–363, 2001.

[19] Z. Zhao, W. Dong, J. Bu, Y. Gu, and C. Chen, "Link-correlation-aware data dissemination in wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5747–5757, 2015.

[20] X. Zheng, J. Wang, W. Dong, Y. He, and Y. Liu, "Bulk data dissemination in wireless sensor networks: analysis, implications and improvement," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1428–1439, 2016.

[21] A. Hagedorn, D. Starobinski et al., "Rateless deluge: over-the-air programming of wireless sensor networks," in *Proceedings of the IEEE International Conference on Information Processing in Sensor Networks (IPSN '08)*, Proceeding of ACM, April 2008.

[22] Y. Hu, M. Dong, K. Ota, A. Liu, and M. Guo, "Mobile target detection in wireless sensor networks with adjustable sensing frequency," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1160–1171, 2016.

[23] X. Zheng, J. Wang, W. Dong, Y. He, and Y. Liu, "Survival of the fittest: data dissemination with selective negotiation in wireless sensor networks," in *Proceedings of the 10th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '13)*, pp. 443–451, IEEE, Hangzhou, China, October 2013.

[24] Q. Yang, S. He, J. Li et al., "Energy-efficient probabilistic area coverage in wireless sensor," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 367–377, 2015.

[25] Y. Liu, A. Liu, and Y. Hu, "FFSC: an energy efficiency communications approach for delay minimizing in internet of things," *IEEE Access*, vol. 4, pp. 3775–3793, 2016.

[26] S. S. Kulkarni and L. Wang, "MNP: multihop network reprogramming service for sensor networks," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, June 2005.

[27] R. Panta, S. Bagchi, I. Khalil et al., "Single versus multi-hop wireless reprogramming in sensor networks," in *Proceedings of the 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM '08)*, Innsbruck, Austria, 2008.

[28] Q. Zhang and A. Liu, "An unequal redundancy level-based mechanism for reliable data collection in wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, article 258, 2016.

[29] H. Cheng, N. Xiong, X. Huang, and L. T. Yang, "An efficient scheduling model for broadcasting in wireless sensor networks," in *Proceedings of the IEEE 27th International Parallel and Distributed Processing Symposium Workshops and PhD Forum (IPDPSW '13)*, pp. 1417–1422, May 2013.

[30] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Proceedings of the 14th Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pp. 517–526, 2004.

[31] Z. Tang, A. Liu, Z. Li, Y.-J. Choi, H. Sekiya, and J. Li, "A trust-based model for security cooperating in vehicular cloud computing," *Mobile Information Systems*, vol. 2016, Article ID 9083608, 22 pages, 2016.

[32] R. Xie, A. Liu, and J. Gao, "A residual energy aware schedule scheme for WSNs employing adjustable awake/sleep duty cycle," *Wireless Personal Communications*, vol. 90, no. 4, pp. 1859–1887, 2016.

[33] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, Upper Saddle River, NJ, USA, 2001.

[34] M. Perillo, Z. Cheng, and W. Heinzelman, "On the problem of unbalanced load distribution in wireless sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference Workshops (GLOBECOM '04)*, pp. 74–79, December 2004.

[35] L. T. Yang, A. B. Waluyo, J. Ma, L. Tan, and B. Srinivasan, *Energy-Efficient Pattern Recognition for Wireless Sensor Networks*, John Wiley & Sons, Hoboken, NJ, USA, 2010.

[36] S. He, J. Chen, and F. Jiang, "Energy provisioning in wireless rechargeable sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 1931–1942, 2013.

[37] Y. Liu, A. Liu, and Z. Chen, "Analysis and improvement of send-and-wait automatic repeat-request protocols for wireless sensor networks," *Wireless Personal Communications*, vol. 81, no. 3, pp. 923–959, 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 6024928, 12 pages
http://dx.doi.org/10.1155/2016/6024928

Research Article

Channel Selection Policy in Multi-SU and Multi-PU Cognitive Radio Networks with Energy Harvesting for Internet of Everything

Feng Hu,[1,2] Bing Chen,[1,2] Xiangping Zhai,[1,2] and Chunsheng Zhu[3]

[1] *The College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China*
[2] *The Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210023, China*
[3] *The Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada V6T 1Z4*

Correspondence should be addressed to Bing Chen; cb_china@nuaa.edu.cn

Received 22 September 2016; Accepted 17 November 2016

Academic Editor: Beniamino Di Martino

Cognitive radio, which will become a fundamental part of the *Internet of Everything* (IoE), has been identified as a promising solution for the spectrum scarcity. In a multi-SU and multi-PU cognitive radio network, selecting channels is a fundamental problem due to the channel competition among secondary users (SUs) and packet collision between SUs and primary users (PUs). In this paper, we adopt cooperative sensing method to avoid the packet collision between SUs and PUs and focus on how to collect the spectrum sensing data of SUs for cooperative sensing. In order to reduce the channel competition among SUs, we first consider the *hybrid* transmission model for single SU where a SU can opportunistically access both idle channels operating either the *Overlay* or the *Underlay* model and the busy channels by using the energy harvesting technology. Then we propose a competitive set based channel selection policy for multi-SU where all SUs competing for data transmission or energy harvesting in the same channel will form a competitive set. Extensive simulations show that the proposed cooperative sensing method and the channel selection policy outperform previous solutions in terms of false alarm, average throughput, average waiting time, and energy harvesting efficiency of SUs.

1. Introduction

Due to the continuous development of wireless devices and services, our environment is transforming into an *Internet of Everything* (IoE) [1–5]. In this IoE paradigm, where everything and everyone will be connected, the bandwidth demand for limited spectrum has been greatly increasing. The scarcity of the spectrum resources has become a serious problem. This is mainly due to the traditional static spectrum allocation policy, where a particular portion of the spectrum can be only used by licensed wireless communications systems. The impoverishment of available spectrum and the underutilization of licensed spectrum facilitate the appearance of cognitive radio (CR) technology, which has evoked much enthusiasm of many scholars, and Federal Communications Commission (FCC) approved unlicensed use of licensed spectrum through CR technology [6–9]. CR has been regarded as an efficient approach to cope up with the spectrum shortage and low utilization problems [10–12]. Therefore, the introduction of cognitive radio in IoE environment can provide on-demand spectrum access among multiple devices.

Dynamic Spectrum Access (DSA) mechanism has been offered for spectrum usage. There are two major transmission models for a secondary user (SU) efficiently using idle spectrums, which are *Overlay* [13] and *Underlay* [14], respectively. In the *Overlay* model, a SU can exclusively and opportunistically use the licensed spectrum only if a primary user (PU) is inactive. In other words, the SU is not allowed to access the spectrum simultaneously with the PU in order to prevent colliding with PU transmission. In contrast, even if when the PU accesses its spectrum, a SU may coexist with it as long as the interference caused to the PU by this SU does not degrade its communication quality in the

Underlay model. However, when the PU state changes to be inactive, the transmission power of the SU will be still below the *interference threshold* constraint in the *Underlay* model. Therefore, the idle spectrum resources are not fully utilized, and the SU does not achieve optimal performance. On the other hand, when the licensed channels are very busy, the time that SU must wait for an available channel is too long, and then it may also significantly reduce the performance of *Overlay* model [15]. Therefore, we need to find a *hybrid* transmission model where the advantages of both *Overlay* and *Underlay* models are combined for the PU state variability, so that the performance of SUs can be maximized.

The *hybrid* transmission model has recently been proposed in [16–18]. In [16], the SU can exchange control information in the *Underlay* model and transmit data information in the *Overlay* model. However, the decision of accessing a model is not based on the sensing results. In [17, 18], the SU can constantly sense the activity of PUs and transmit data information in the *Overlay* model when the PU transmission is not detected. Otherwise, the SU reduces its transmission power to access the spectrum in the *Underlay* model. However, these papers did not take into account the sensing errors and neglected the effect of PU retransmission on the SU QoS. Although these related works indicated that the SU can obtain more spectrum access opportunities in the *hybrid* transmission model compared with the two conventional transmission models, the issue of two or more SUs competing for the same channel has rarely been studied so far to the best of our knowledge. Furthermore, the packet collision probability between SUs and a PU will increase in the multiple SUs scenario [19]. Thus, due to the importance of the collision avoidance in a CR network with multiple SUs and multiple PU channels, we need to propose a channel selection policy in the *hybrid* transmission model to address the collision issue.

Energy supply is always a critical issue in wireless communications. In a multi-SU and multi-PU CR network where multiple SUs access multiple PU channels in the *hybrid* transmission model, the SU needs to spend more energy to constantly detect many channels and switch among multiple channels. Therefore, energy efficiency is another important criterion in the CR network along with spectrum efficiency [20, 21]. Furthermore, the cost of replacing the battery is often expensive. Recently, some energy harvesting techniques have been introduced in [22–24]. Such techniques allow devices to harvest natural sources' energy such as sun, wind, acoustic, and ambient radio frequency (RF) waves. Converting electromagnetic waves from ambient RF waves into energy is considered to be more suitable and stable for the low energy devices in sensor networks or CR networks compared with other sources [24]. Assuming that a SU is equipped with the RF energy harvesting capability, it must not only select an idle channel to transmit data but also a busy channel to harvest RF energy to obtain enough energy and spectrum usage opportunity. Hence, a suitable channel selection policy is very important to improve both the spectrum efficiency and the energy efficiency in CR networks.

Inspired by the inherent benefits of the above schemes, in this paper, we focus on the channel competition among SUs and packet collision between SUs and PUs in multi-SU and multi-PU CR networks. Apart from the existing works, such as adopting the conventional noncooperative spectrum sensing method in the multi-SU CR network [25], and allowing SUs to access idle channels in the *Overlay* or *Underlay* model [14], there is no effective solution to the packet competition among multiple SUs and PUs [26]. We adopt the cooperative sensing method and the concept of competitive set to solve these two problems so that the spectrum sensing accuracy and the throughput of multiple SUs can be improved. It is noted that, in our study, SUs can harvest RF energy from busy channels by using the energy harvesting technology so as to extend their battery life.

The main contributions of this paper are as follows:

(i) We use the cooperative sensing method to avoid the packet collision between SUs and PUs. In channel sensing phase, the SUs that detect the same channel exchange the channel usage information with each other and make a more accurate decision on the state of this channel. Moreover, the packet collision between SUs and PUs can significantly decrease, since the cooperative sensing method can detect the activity of PUs reoccupy their channels with a large probability.

(ii) We propose a *hybrid* transmission model combining *Overlay* and *Underlay* models to fully utilize the available idle spectrum. Each SU can opportunistically access the unoccupied PU channel or underlay part of its signal into the portion of the channel occupied by the PU depending on the data queue state and sensing result or decide whether or not to access a busy channel to harvest RF energy given its energy queue. Extensive simulations show that our proposed *hybrid* transmission model can improve the efficiency of spectrum usage and the energy harvesting efficiency of SUs.

(iii) With the aim of eliminating the channel competition among SUs and reducing their average waiting time and also decreasing their spectrum handoff delay, we propose a competitive set based channel selection policy. In our proposed policy, the SUs who form a competitive set in the same channel randomly obtain integer labels from *zero*. The SU who obtains the *zero* label has the right to use this channel. In particular, several SUs can obtain multiple labels in different competitive sets. Therefore, the average waiting time that SUs spend on switching to other idle channels or still staying on this channel for data transmission or accessing a busy channel for energy harvesting is lower compared with the random selection policy. Simulation results show that the proposed channel selection policy is simple and effective on reducing the collisions among SUs.

The rest of this paper is organized as follows. The system model is described in Section 2, and the cooperative spectrum

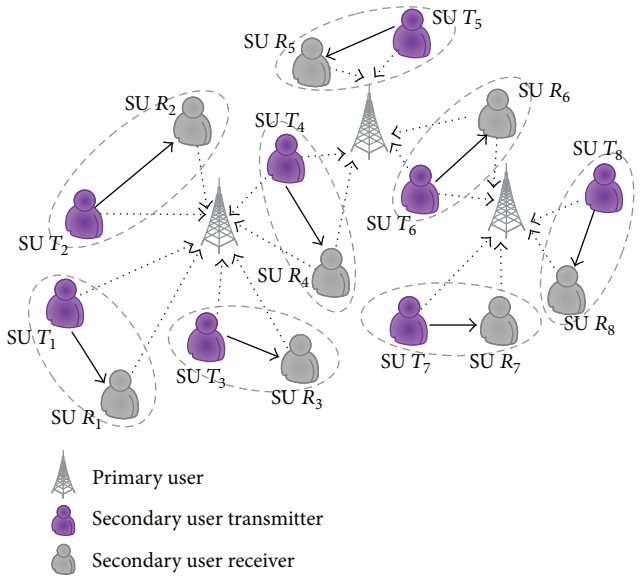


Figure 1: A scenario of the multi-SU and multi-PU CR network.

sensing method is given in Section 3. In Section 4, we present the *hybrid* transmission model and channel selection policy and then analyze the performance in terms of average throughput, average waiting time, and energy harvesting efficiency of SUs' three aspects. The simulation results are listed in Section 5. Finally, we conclude our work in Section 6.

2. System Model

2.1. Multi-SU and Multi-PU CR Networks. We consider a multi-SU and multi-PU CR network with M PUs and N pairs of SU, as depicted in Figure 1, where each PU is allocated a licensed channel (which we call "PU channel"). Similar to [13–15], the traffic of each channel is modeled as a two-state continuous-time Markov process: the spectrum is occupied by the PU (*busy* state) and the spectrum is not occupied by the PU (*idle* state). For the PUs, these two states are referred to as ON and OFF states, respectively. Each SU transmitter and its corresponding SU receiver are within each other's transmission range. Therefore, the existence of a communication between two SUs depends not only on the distance between them, but also on the time-varying activities of the PUs. As illustrated in Figure 1, we consider the scenario that several SUs may access the same channel, and one SU may have more than one channel for selection.

As these PUs are in the interference range of some SUs, the channel power gains from the PU transmitter to the PU receiver, SU transmitter to the SU receiver, PU transmitter to the SU receiver, and SU transmitter to the PU receiver are denoted by G_{pp}, G_{ss}, G_{ps}, and G_{sp}, respectively. We employ the model $G_{ij} = kd_{ij}^{-\alpha}$ for the channel gain between the ith transmitter and the jth receiver, where k is an attenuation factor that represents power variation caused by path loss, d_{ij} denotes the distance between them, and α is the path loss [27]. We assume that the channel power gains and the channel state information (CSI) are known to each SU, and SUs can obtain the channel availability after spectrum sensing.

2.2. Energy Harvesting Technology. RF energy signal can not only propagate over a distance but also broadcast in all directions [22]. However, due to the uncertainty of location, fading, and environmental conditions, the energy supplied from RF energy may not guarantee QoS in wireless applications. To ensure the static and stable energy, the RF energy signal is transformed to a DC voltage and then stored into a rechargeable battery [23]. It is reasonable to define the effective zone of the energy harvesting, since the propagation energy drops off rapidly with the distance increases. We assume that each SU can only obtain the RF energy signal from the channels that it can sense. Each SU can harvest RF energy from the busy channels occupied by PUs and store the energy in a rechargeable battery when its transmitter is equipped with an energy harvesting device, and the maximum size of battery is $E_{\max}$. In this paper, the rechargeable battery is modeled by an ideal linear model [28], where the changes in the energy stored are linearly related to the amounts of energy harvested or spent. Since the increased energy harvested from PU channels can be utilized for channel sensing and data transmission, the working time of SUs will be extended.

3. Cooperative Spectrum Sensing

Spectrum sensing is the basis of the DSA mechanism. Furthermore, sensing errors will affect the performance of SUs transmission and cause packet collision between SUs and PUs. In this section, we describe our cooperative spectrum sensing method.

3.1. Energy Based Spectrum Sensing. Spectrum sensing has to be performed before data transmission to detect the channel availability. Many signal techniques have been used for the SUs to sense the activity of the PUs [13]. The energy detection method not only implements simply but also represents intuitively the proportion of the busy channels. Therefore, the energy detection method is accurate and optimal when the SUs have little or no prior knowledge of the PU signal [29], and we consider it as the spectrum sensing algorithm in our proposed policy. The aim of spectrum sensing is to sense the existence of signal in licensed spectrum. Thus, under the two hypotheses, the signal can be expressed as

$$\begin{aligned} H_0 &: x(t) = n(t), \\ H_1 &: x(t) = s(t) + n(t), \end{aligned} \tag{1}$$

where $n(t)$ is an Additive White Gaussian Noise (AWGN) and $s(t)$ is the signal of PU in target channel. H_0 and H_1 are the two hypotheses of nonexistence or existence of $s(t)$. From [30], we have known that the probability of detection can be denoted by P_d with a fixed SNR γ in an AWGN channel, and it can be written as

$$P_d(\gamma, \tau, \lambda) = \mathcal{Q}\left(\left(\frac{\lambda}{\sigma^2} - \gamma - 1\right)\sqrt{\frac{\tau f_s}{2\gamma + 1}}\right), \tag{2}$$

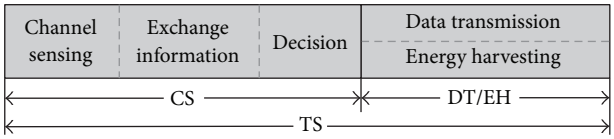


Figure 2: An intuitional illustration of the time-slot structure.

where τ is the sensing duration, λ is the sensing threshold, f_s is the sampling frequency, σ^2 is the variance of the AWGN, and $\mathcal{Q}(x)$ is the tail probability of the normal distribution. Under imperfect sensing, there are two types of sensing errors: *miss detection* and *false alarm*. A *false alarm* error occurs when the SU observes the channel is busy whereas it is actually idle, and a *miss detection* error occurs when the SU observes the channel is idle whereas it is actually busy. Hence, the *false alarm* indicates the waste of spectrum access opportunity, whereas the *miss detection* imposes on the potential interference to PUs. The *false alarm* probability P_f and *miss detection* probability P_m can be expressed as [31]

$$P_f(\tau, \lambda) = \mathcal{Q}\left(\left(\frac{\lambda}{\sigma^2} - 1\right)\sqrt{\tau f_s}\right),$$
$$P_m(\gamma, \tau, \lambda) = 1 - \mathcal{Q}\left(\frac{\lambda/\sigma^2 - (1+\gamma)}{(1+\gamma)\sqrt{2/\tau f_s}}\right), \quad (3)$$

where P_f and P_m are related to the threshold λ and the sensing time τ. Furthermore, P_m is also a function of SNR.

3.2. Cooperative Spectrum Sensing. Due to the effects of multipath fading, inside buildings with high penetration loss and local interference, the probability of *miss detection* and *false alarm* will be increased under the conventional non-cooperative spectrum sensing method. This phenomenon will lead to packet collision between SUs and PUs in multi-SU and multi-PU CR networks. In order to deal with this problem, cooperative spectrum sensing has been adopted in some studies [15, 17, 23]. We focus on how to collect the spectrum sensing data of SUs for cooperative sensing and combine these sensing results to produce the final decision in this paper.

As illustrated in Figure 2, we suppose the multi-SU and multi-PU CR network with time slotted (TS), that is, one TS consists of two phases, which are the channel sensing phase (CS) and the data transmission (DT) or energy harvesting (EH) phase, respectively. In the first phase, SUs sense the PU channels to detect the activity of the PUs and exchange the channel usage information with other SUs. Then, each SU will combine its sensing results with others'. At last, two or more of the same results are considered to be the final decision of this channel. In particular, the channel will be redetected until a decision is made when a channel is detected by four SUs, and two of them believe that the channel is idle while the other two are just the opposite. The sensing result is considered to be the final decision when a channel is only detected by one SU. In the next phase, the SU executes RF energy harvesting or data transmission based on the final decision. Similar to [32, 33], we suppose that sensing duration is small, compared with the PU channel traffic state cycle, so that the PU channel traffic state can be considered unchanged during sensing phases.

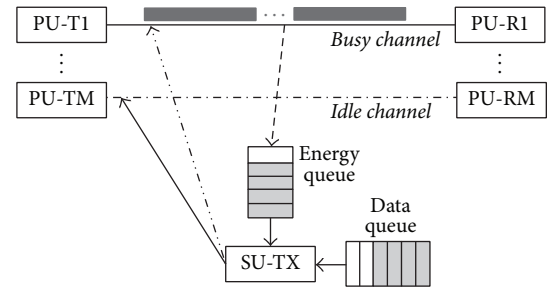


Figure 3: An illustration of the *hybrid* transmission model.

4. Channel Selection Policy

In this section, we first propose a *hybrid* transmission model for single SU and secondly present a channel selection policy for multi-SU based on the competitive set to alleviate the channel competition among SUs.

4.1. Hybrid Transmission Model. As shown in Figure 3, the arriving data is buffered in the data queue of the SU transmitter, Q_{Di}, $i = 1, 2, 3, \ldots, N$. The maximum capacity of the data queue is $Q_{\max}$. As mentioned before, the RF energy is stored in the energy queue, Q_{Ei}, $i = 1, 2, 3, \ldots, N$, whose maximum size is denoted as $E_{\max}$.

At the beginning, when the data arrives at ith SU transmitter, its data queue and energy queue can be represented as $Q_{Di} \neq \varnothing$, $Q_{Ei} = E_{\max}$; then the SU can perform data transmission when idle channels are sensed. Let $E(s)$ be the sensing outcome of ith SU, and we define λ_O and λ_U be the *Overlay* and the *Underlay* model energy threshold, respectively. If the channel signal energy is sensed below the *Overlay* model energy threshold, that is, $E(s) < \lambda_O$, the SU will transmit data with a higher power from its data queue in the *Overlay* model. However, if the channel signal energy is above the *Overlay* model energy threshold but below the *Underlay* model energy threshold, that is, $\lambda_O < E(s) < \lambda_U$, it means that the PU does not fully occupy this channel, and the SU can access it with PU at the same time by reducing its transmission power as long as it does not interfere in the PU transmission, that is, *Underlay* transmission model.

To make use of the *hybrid* transmission model, each SU transmitter is assumed to have perfect knowledge of the CSI. For different channels, their capacity and utilization rate are different. Based on the sensing result of channels, each SU calculates the statistical *Overlay* and *Underlay* model energy threshold and updates them according to the two types of sensing errors, that is, *miss detection* and *false alarm*. When the data arrives at a SU transmitter, it compares the current channel sensing results with the knowledge of CSI to obtain the occupancy of PU. The SU estimates the power of the PU based on the transmission distance and antenna gain when the PU does not fully occupy this channel [34]. For

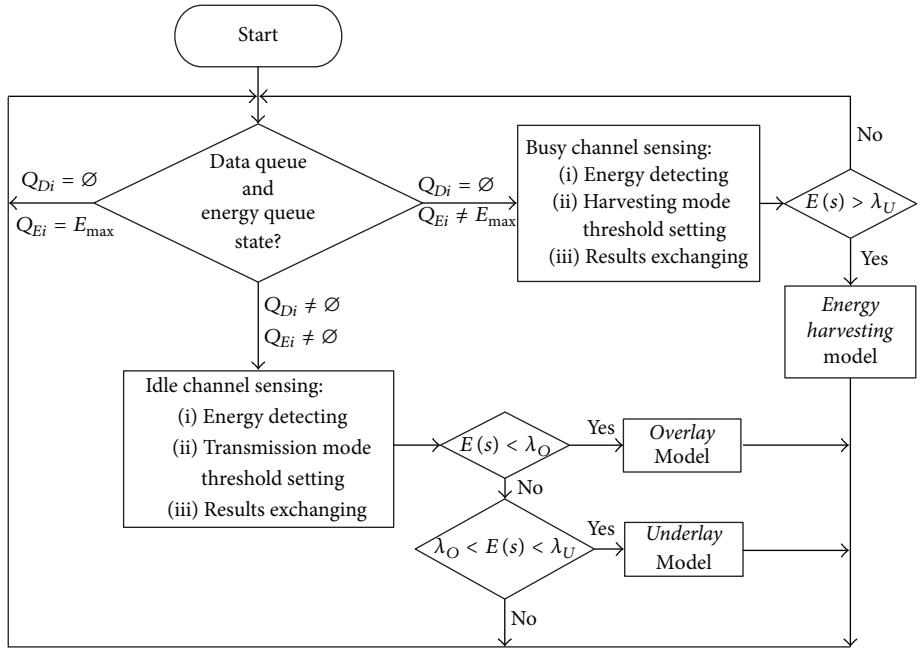


Figure 4: An illustration of the process of selecting the transmission models for each SU.

the *Overlay* model, there is no limitation on the transmit power of SUs, and they can transmit data with the initial power. Nevertheless, due to the interference caused by SU towards PU, the SUs need to decrease the transmit power, change the modulation type, and adjust the encoded mode to afford a suitable SNR to accommodate the variation of current channel in the *Underlay* model.

When the data queue of ith SU is empty and its energy is used in the previous time-slot, that is, $Q_{Di} = \varnothing$, $Q_{Ei} \neq E_{\max}$, the SU can harvest RF energy from busy channels for increasing energy reserves. Hence, if a channel is sensed above the *Underlay* model energy threshold, the SU may implement energy harvesting from it.

In our proposed *hybrid* transmission model, each SU can determine to implement either the data transmission or the energy harvesting depending on the state of data queue and energy queue. Based on the spectrum sensing results and *Overlay*/*Underlay* model energy thresholds, each SU can not only access a channel alone or with the PU simultaneously for data transmission but also harvest the RF energy from the PU occupied channels. The SU decides whether or not to stay in the current channel or switch to a new channel for data transmission or a busy channel for energy harvesting after sensing channels in the next CS phase. The process of selecting the transmission models for each SU is presented in Figure 4.

4.2. Overview of Channel Selection Policy. As the mentioned *hybrid* transmission model, each SU can implement either the data transmission in an idle channel or the energy harvesting from a busy channel. However, for the multi-SU and multi-PU CR network, one of the great challenges of implementing multi-SU channel access successfully is the problem of competition among SUs. We explain the details of the proposed channel selection policy.

4.2.1. Channel Selection Policy for Data Transmission. The SU transmitter sends a RTS packet on the channel to its corresponding SU receiver if an idle channel is detected. Then the SU receiver replies with a CTS packet in the same channel. Notice that the RTS/CTS collision may occur when more than one pair of SUs contends the same target idle channel for data transmissions. Hence, different from the conventional way, the SU pair does not access the idle channel immediately when the CTS packet is successfully received by the SU transmitter. Those SUs who receive CTS packet form a competitive set, S_{Ii}, $i = 1, 2, 3, \ldots, M$, which means that these SUs are competing to access this PU channel. Supposing that the size of S_{Ii} is W, we randomly assign them integer labels from *zero* to $W - 1$. The SU who obtains the *zero* label can transmit data in the DT phase. In particular, for the SUs who can sense more than one channel, they can compete for multiple idle channels and obtain multiple labels when the data arrives at their data queue. Furthermore, the SU can access the corresponding channel for data transmission as long as it can obtain the *zero* label in one competitive set. Similarly, when the channel can only be accessed in the *Underlay* model for data transmission, those SUs who receive the CTS packet will form a competitive set, S_{Ui}, $i = 1, 2, 3, \ldots, M$.

The SU that transmits data in the previous time-slot will keep data transmission until the channel state is changed when the sensing outcome of the current channel is $E(s) < \lambda_O$ in the next CS phase. All the label values of other SUs are in the same competitive set minus one when the SU withdraws from the current channel. Therefore, the SU whose

Input: All PUs channels, l_n, $n \in [1, N]$, including l_m idle channels, $m \in [1, M]$, and p SUs, $p \in [1, P]$.
Output: Channel selection for SUs

```
(1)  begin
(2)     /* For l_m idle channels */
(3)     if the data queue of SUs, Q_Dp ≠ ∅ then
(4)        if l_m ≥ 1 then
(5)           if the q SUs receive CTS packet successfully then
(6)              The q SUs form a number of competitive sets, S_Ii, i = 1, ..., m;
(7)              Randomly assigning them integer labels from zero to w − 1, w ≤ q;
(8)              for erery competitive sets, S_Ii do
(9)                 The SU who obtains the zero label can access the corresponding channel for data transmission, and
                    withdraw from other competitive sets;
(10)                Label values of other SUs minus one;
(11)             end
(12)          end
(13)       end
(14)       if l_m = 0 then
(15)          if the s SUs receive CTS packet successfully then
(16)             if the t SUs transmit the data in the Underlay model, and their throughput can be satisfied then
(17)                The t SUs form a number of competitive sets, S_Ui, i = 1, ..., n;
(18)                Randomly assigning them integer labels from zero to v − 1, v ≤ t;
(19)                for erery competitive sets, S_Ui do
(20)                   The SU who obtains the zero label can reduce its transmitting power and access the corresponding
                       channel for data transmission, and then withdraw from other competitive sets;
(21)                   Label values of other SUs minus one;
(22)                end
(23)             end
(24)          end
(25)       end
(26)    end
(27) end
```

Algorithm 1: The channel selection policy for data transmission.

label value is subtracted to be *zero* can access this channel. If the present channel is detected satisfy $\lambda_O < E(s) < \lambda_U$ in the next CS phase; that is, the PU is not completely occupied in this channel for data transmission, the SU reduces its transmission power to satisfy the interference power constraint of PU that it can continue to transmit data in the *Underlay* model, and their corresponding competitive sets will remain in use. However, the transmission of SU will cause interference to the communication of PU if the $E(s) > \lambda_U$. Then the data transmission of SU will be stopped in the next DT phase, and the competitive set of this channel will be dissolved. The algorithm of our channel selection strategy for data transmission is presented in Algorithm 1.

We illustrate the process of randomly assigning integer labels by Figure 5. First, four SUs need to access channels for data transmission, and they sense the current channel availability to obtain a list of available channels. Secondly, the SUs who compete for the same channel form a competitive set; that is, the SUs 1, 2, 3 can use the channel A. We randomly assign integer labels from *zero* for these SUs. Thirdly, the SUs who obtain the *zero* label can access the channels, and they withdraw from the corresponding competitive sets while the label values of other SUs are minus one. In particular, the SU 2 can use channels A or B. In the next time-slot, the SU 4 can access the channel B.

4.2.2. Channel Selection Policy for Energy Harvesting. The SUs that contend for the same target busy channel form a competitive set, S_{Ej}, $j = 1, 2, 3, \ldots, M$, which means that these SUs are competing to access the jth PU channel for energy harvesting. We also assign them integer labels, and the SU that obtains the *zero* label can harvest energy in the next EH phase. Then these SUs withdraw from the competition sets when the data arrives at the data queue of SUs or their energy queue is full. The algorithm of our channel selection policy for energy harvesting is given in Algorithm 2.

In the proposed channel selection policy, the SU receiver sends a Decode packet to its transmitter when the current transmission is complete; that is, the data has been accepted successfully. The Channel-Switching (CSW) flag is set when the SU needs to switch to another channel, and then the SU transmitter and receiver pause their current transmission and perform channel handoff [35, 36].

Note that the CSMA/CA protocol also uses the RTS/CTS handshake procedures to ensure that the collision does not occur among users and utilizes the exponential-backoff algorithm to decompose collision; that is, each node performs a random delay t when the collision happens, and t obeys the $T(0 \sim T)$ on the bottom of the exponential distribution. In our proposed channel selection policy, we ensure the usage of idle channels through establishing the competitive sets

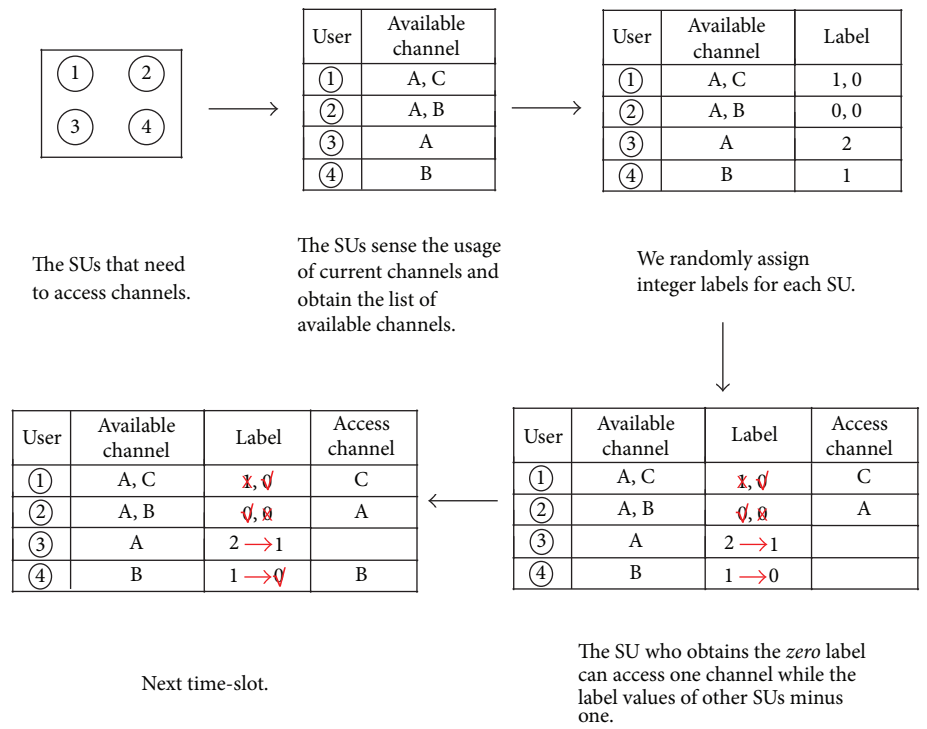


Figure 5: An example of randomly assigning integer labels.

Input: All PUs channels, l_n, $n \in [1, N]$, including l_m idle channels, $m \in [1, M]$, and p SUs, $p \in [1, P]$.
Output: Channel selection for SUs

```
(1)  begin
(2)    /* For l_{n-m} busy channels */
(3)    if the data queue of SUs, Q_{Dp} = ∅ then
(4)      if the energy queue of SUs, Q_{Dp'} ≠ Q_max then
(5)        The p' SUs form a number of competitive sets, S_{Ei}, i = 1, ..., n;
(6)        Randomly assigning them integer labels from zero to u − 1, u ≤ p';
(7)        for erery competitive sets, S_{Ei} do
(8)          The SU who obtains the zero laber can access the corresponding busy channel for energy harvesting in the next
             EH phase, and withdraw from other competitive sets;
(9)          Label values of other SUs minus one;
(10)       end
(11)     end
(12)   end
(13) end
```

Algorithm 2: The channel selection policy for energy harvesting.

and randomly assigning the integer labels when the collision occurs. Moreover, in order to reduce the average waiting time of SUs, the SUs who access channels withdraw from their competitive sets while other SUs' label values are minus one.

4.3. Performance Analysis of the Channel Selection Policy. In this section, we intend to illustrate the spectrum usage performance of our proposed policy in terms of average throughput, average waiting time, and energy harvesting efficiency of SUs' three aspects.

4.3.1. Average Throughput of SUs. In our proposed channel selection policy, SU can transmit data in the *Overlay* or *Underlay* model. The service rate of each SU in the *hybrid* model is described as $R_h = R_o + R_u$, and R_o can be denoted by R_o^0, R_o^1, $R_o^{0'}$, and $R_o^{1'}$ in the *Overlay* model [37]

$$R_o^0 = B \log_2 \left(1 + g_s P_s^o\right),$$

$$R_o^1 = 0,$$

$$R_o^{0'} = B\log_2\left(1 + \frac{g_s P_s^o}{g_p P_p + 1}\right),$$
$$R_o^{1'} = 0, \tag{4}$$

where R_o^0 represents that the PU does not occupy the channel. In contrast, R_o^1 represents that the spectrum is being occupied by PU. $R_o^{0'}$ and $R_o^{1'}$ are the service rate of each SU under *false alarm* and *miss detection*, respectively. Similarly, the R_u can be denoted by R_u^0, R_u^1 in the *Underlay* model

$$R_u^0 = B\log_2\left(1 + g_s P_s^u\right),$$
$$R_u^1 = B\log_2\left(1 + \frac{g_s P_s^u}{g_p P_p + 1}\right). \tag{5}$$

The throughput of SU can be described in terms of the outage as [38]

$$T = 1 - p_{\text{out}}, \tag{6}$$

where p_{out} is the outage probability. The throughput in channel selection policy, T_h, is comprised of T_o and T_u, respectively. T_o is given by

$$T_o = p_i\left(1 - p_f\right)\left(1 - p_{\text{out}}^o\right) + \left(1 - p_i\right)p_f\left(1 - p_{\text{out}}^{o'}\right), \tag{7}$$

where p_i is the channel idle probability. Since the SU transmission will cause interference to PU Under *false alarm*, p_{out}^o and $p_{\text{out}}^{o'}$ can be described as

$$p_{\text{out}}^o = \Pr\left[R_o^0 < R_s\right],$$
$$p_{\text{out}}^{o'} = \Pr\left[R_o^{0'} < R_s\right], \tag{8}$$

where R_s is the required service rate of SU. Correspondingly, we can obtain T_u, p_{out}^u, and $p_{\text{out}}^{u'}$ as follows:

$$T_u = p_i\left(1 - p_f\right)\left(1 - p_{\text{out}}^u\right) + \left(1 - p_i\right)p_f\left(1 - p_{\text{out}}^{u'}\right),$$
$$p_{\text{out}}^u = \Pr\left[R_u^0 < R_s\right], \tag{9}$$
$$p_{\text{out}}^{u'} = \Pr\left[R_u^1 < R_s\right].$$

4.3.2. Average Waiting Time of SUs. Here, we calculate the time elapsed between each SU which receives the RTS signal and implements data transmission, and this elapsed time can reflect the performance of the competitive set. The average waiting time of SUs will be longer than the conventional random access policy if the design of the competitive set is not reasonable. Thus, the average waiting time of SUs, T_w, can be described as

$$T_w = T_t - T_{\text{RTS}}, \tag{10}$$

where T_t and T_{RTS} are the time-slots that the SU transmits data to and receives the RTS signal from, respectively.

Table 1: Simulation parameters settings.

Parameter	Value
P_i	0.8
λ_o	0.3
λ_u	0.7
$E_{\max}$	15
$Q_{\max}$	20
R_s	3 bps
P_p	15 dB
Time-slot	2 ms
RTS packets length	250 bit
CTS packets length	220 bit

4.3.3. Energy Harvesting Efficiency. We use e_h to express the packets of energy that can be harvested by SUs from busy channels, and it follows Poisson distribution. The energy consumed by SU for data transmission and spectrum sensing are e_t and e_s, respectively. We assume that e_c represents another energy consumption on circuit and e_r^t denotes the residual energy at the time-slot t. Therefore, the energy harvesting efficiency is described in terms of the residual energy in the next time-slot

$$e_r^{t+1} = \min\left[e_r^t + e_h - \left(e_t + e_s + e_c\right), E_{\max}\right]. \tag{11}$$

5. Simulations

In this section, we will provide numerical results to demonstrate the performance of the proposed cooperative sensing method and channel selection policy in terms of probability of false alarm, average throughput, average waiting time, and energy harvesting efficiency of SUs. Table 1 shows the parameter settings of our simulations, and some parameters are valued based on the previous works on CR networks. We consider a multi-SU and multi-PU CR network with 20 PUs, 20 available PU channels, and 25 pairs of SUs. In particular, several SUs may access the same channel, and one SU may have more than one channel for selection. We set all the spectrum bandwidth to be the same, and the packets lengths of SUs and PUs are fixed in the simulations. However, the interference limitation of PUs is different. We let the path loss constant $\alpha = 2$ and attenuation factor $k = 0.5$, respectively, according to the empirical values to compute the channel gain. Moreover, the white Gaussian noise is 8×10^{-15}.

5.1. Performance for Probability of False Alarm. Figure 6 illustrates the probability of false alarm in cooperative sensing and conventional noncooperative sensing method under different numbers of SUs. As shown in the figure, the probability of false alarm decreases with the increase of detection probability. For the conventional noncooperative sensing method, there is no interaction on the detection results among multiple SUs. Hence, the increase of users has no impact on the probability of false alarm. For a fixed detection probability, cooperative sensing method can achieve higher detection accuracy. As described in Section 3.2, two or more

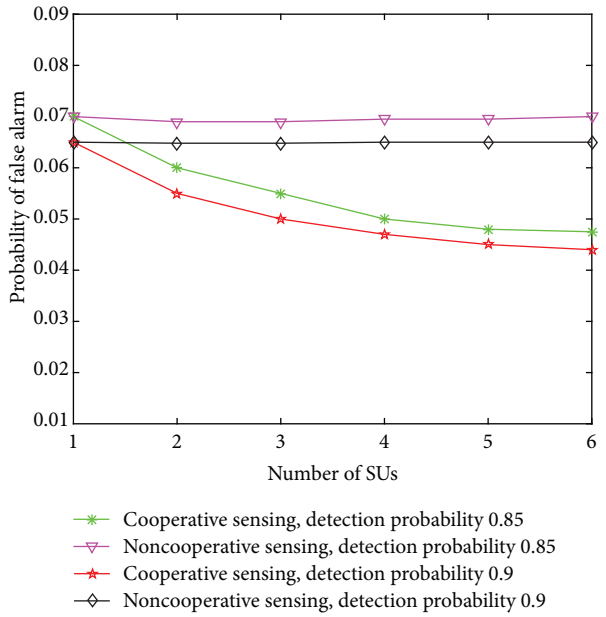


FIGURE 6: The probability of false alarm in cooperative sensing and noncooperative sensing methods under different numbers of SUs.

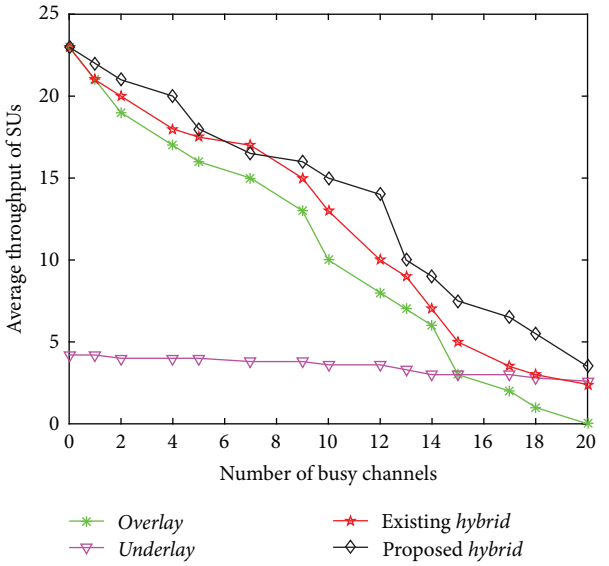


FIGURE 7: The effect of transmission models on the average throughput of SUs under different numbers of busy channels.

of the same results are considered to be the final decision of the target channel. Therefore, the probability of false alarm decreases significantly when the cooperative SUs are 2 or 3. Furthermore, the probability of false alarm where the detection probability is 0.85 decreases faster when the number of SUs changes from 3 to 4. Thus, the cooperative sensing method can improve the accuracy of channel sensing in the case of low detection rate. However, more than 4 cooperative SUs have little effect on the probability of false alarm.

5.2. Performance for Average Throughput of SUs. Figure 7 shows the average throughput of SUs in the following transmission models: our proposed *hybrid* model, existing *hybrid* model [39], *Overlay*-only model, and *Underlay*-only model under different numbers of busy channels. As can be seen from the figure that the average throughput of SUs decreases in the *Overlay*-only model with the number of busy channels increase. It is due to the fact that the high percentage of busy channels restricts SUs from transmission in the *Overlay*-only model. However, there is a little impact on the average throughput of SUs, since the SUs can coexist with PUs in the *Underlay*-only model. It can be observed from the figure that the *hybrid* model transmission outperforms the *Overlay*-only and *Underlay*-only model alone. Furthermore, we can see that our proposed *hybrid* model can achieve higher throughput compared with the existing *hybrid* model. The reason of this observation can be explained as follows. With the decrease of available idle channels, the opportunities of SUs for data transmission become less. The collision among SUs becomes more intense, since the existing *hybrid* model is only based on the number of SUs and access probability. However, the concept of competitive set can improve the utilization of the limited idle channels.

5.3. Performance for Average Waiting Time of SUs. Figure 8 shows the average waiting time of SUs in four different models under different numbers of busy channels. It is clear that the average waiting time of SUs greatly increases in the *Overlay*-only model with the decrease of available idle channels. In contrast, the average waiting time is not significantly increased in the *Underlay*-only model, since the number of available idle channels has little effect on the data transmission of SUs. The different *interference threshold* constraint of PUs so that the SUs cannot access some channels is in the *Underlay*-only model, which results in the consequence that some SUs need to wait for a long time to access channels. The average waiting time of SUs of our proposed *hybrid* model is lower than the existing *hybrid* model but higher than the *Underlay*-only model when lots of channels are occupied by PUs. The explanation for this observation is as follows. As described in Section 4, SUs can continue to transmit data in the *Underlay* model when the PU accesses its idle channel, and their current competitive sets will remain in use. Furthermore, the SUs that compete more than one channels can wait for accessing opportunities in other competitive sets when the current channel cannot be accessed. Therefore, the average waiting time of SUs will be reduced in our proposed *hybrid* model. However, since the *hybrid* model will give priority to whether the channel can be accessed in the *Overlay* model when the idle channels become less, the average waiting time of SUs in the *Underlay*-only model is lower than ours.

5.4. Performance for Energy Harvesting Efficiency of SUs. Figure 9 shows the average residual energy of SUs in the conventional CR network, existing CR network with energy harvesting [30], and our CR network with energy harvesting versus the simulation time. As shown in the figure, energy harvesting technology can ensure that enough energy is reserved after long time communication. Furthermore, our

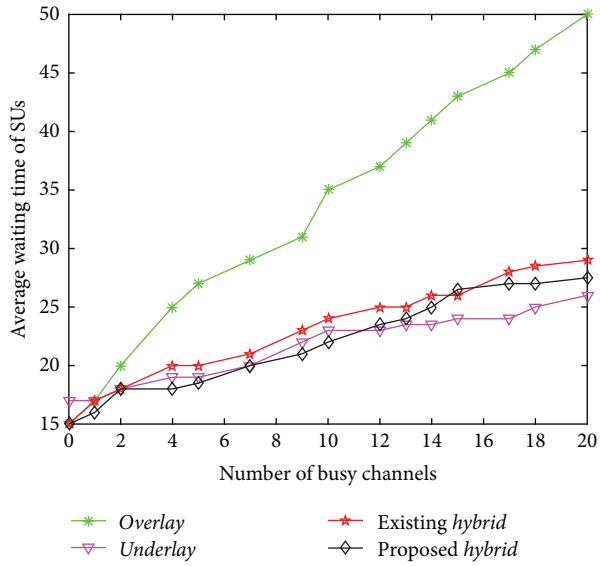


Figure 8: The effect of transmission models on the average waiting time of SUs under different numbers of busy channels.

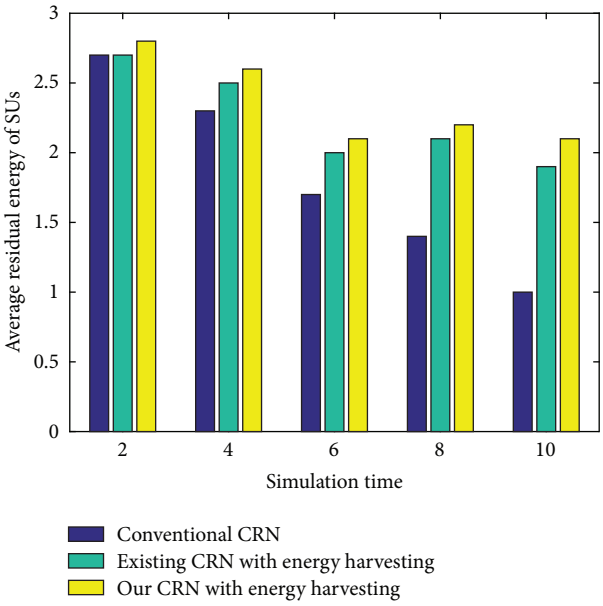


Figure 9: The average residual energy of SUs in the conventional CR network, existing CR network with energy harvesting, and our CR network with energy harvesting under different simulation time.

proposed CR network with energy harvesting outperforms the existing CR network with energy harvesting. This is because as described in Section 3, SUs decide to sense the idle channels for data transmission or busy channels for energy harvesting depending on the state of data queue and energy queue. Hence, SUs can spend less energy for sensing channels. Meanwhile, the SUs may have more opportunities to harvest energy, since the concept of competitive set can reduce the collision among multiple SUs competing for the same busy channel.

6. Conclusion

In this paper, aiming at solving the problem of spectrum scarcity in IoE environment, we consider a multi-SU and multi-PU cognitive radio network in which the SUs are equipped with the RF energy harvesting capability. In this network, the crucial issues are the channel competition among SUs and the packet collision between SUs and PUs. We adopt the cooperative spectrum sensing method to reduce the probability of sensing errors and alleviate the interference to PUs. In order to solve the problem of channel competition among SUs, we first propose a *hybrid* transmission model for single SU. Each SU can either implement data transmission in an idle channel or energy harvesting from a busy channel given its data queue and energy queue state and sensing result. Additionally, we present a channel selection policy for multi-SU based on competitive set. Our proposed policy can achieve higher throughput compared with the conventional random policy. Furthermore, the collision will never be detected by themselves and may last for a quite long time when several SUs collide with each other in the conventional random policy. Hence, the channel competition among SUs will largely limit the performance of conventional random policy. While SUs will detect the collision in the CS phase and stop transmission in the next DT/EH phase to avoid longer ineffective transmission in our proposed policy. Simulations show that the proposed cooperative sensing method and channel selection policy outperform previous solutions in terms of probability of false alarm, average throughput, average waiting time, and energy harvesting efficiency of SUs.

Competing Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China, under Grant 61672283, the Funding of Jiangsu Innovation Program for Graduate Education, under Grant KYLX15 0325, the Fundamental Research Funds for the Central Universities, under Grant NS2015094, the Natural Science Foundation of Jiangsu Province, under Grant BK20140835, and the Postdoctoral Foundation of Jiangsu Province, under Grant 1401018B.

References

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

[2] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a smart city through internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 112–121, 2014.

[3] C. Zhu, V. C. M. Leung, L. Shu, and E. C. H. Ngai, "Green internet of things for smart world," *IEEE Access*, vol. 3, pp. 2151–2162, 2015.

[4] Z. Sheng, C. Mahapatra, C. Zhu, and V. C. M. Leung, "Recent advances in industrial wireless sensor networks toward efficient management in IoT," *IEEE Access*, vol. 3, pp. 622–637, 2015.

[5] Z. Sheng, C. Zhu, and V. C. M. Leung, "Surfing the internet-of-things: lightweight access and control of wireless sensor networks using industrial low power protocols," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 14, no. 1, article e2, pp. 1–11, 2014.

[6] W. Li, C. Zhu, V. C. M. Leung, L. T. Yang, and Y. Ma, "Performance comparison of cognitive radio sensor networks for industrial IoT with different deployment patterns," *IEEE Systems Journal*, 2015.

[7] W. Li, V. Leung, C. Zhu, and Y. Ma, "Scheduling and routing methods for cognitive radio sensor networks in regular topology," *Wireless Communications and Mobile Computing*, vol. 16, no. 1, pp. 47–58, 2016.

[8] J. Li, H. Zhao, J. Wei et al., "Sender-jump receiver-wait: a blind rendezvous algorithm for distributed cognitive radio networks," in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '16)*, Valencia, Spain, September 2016.

[9] Federal Comunications Commission, "Unlicensed operation in the TV broadcast bands," Rep. ET Docket no. 08-260, 2008.

[10] Y. C. Liang, K. C. Chen, G. Y. Li, and P. Mahonen, "Cognitive radio networking and communications: an overview," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3386–3407, 2011.

[11] E. Z. Tragos, S. Zeadally, A. G. Fragkiadakis, and V. A. Siris, "Spectrum assignment in cognitive radio networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1108–1135, 2013.

[12] X. Zhai, L. Zheng, and C. W. Tan, "Energy-infeasibility tradeoff in cognitive radio networks: price-driven spectrum access algorithms," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 3, pp. 528–538, 2014.

[13] Y. Yilmaz, Z. Guo, and X. Wang, "Sequential joint spectrum sensing and channel estimation for dynamic spectrum access," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 11, pp. 2000–2012, 2014.

[14] N. Khambekar, C. M. Spooner, and V. Chaudhary, "On improving serviceability with quantified dynamic spectrum access," in *Proceedings of the IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN '14)*, pp. 553–564, McLean, Va, USA, April 2014.

[15] T. M. C. Chu, H. Phan, and H. J. Zepernick, "Hybrid interweave-underlay spectrum access for cognitive cooperative radio networks," *IEEE Transactions on Communications*, vol. 62, no. 7, pp. 2183–2197, 2014.

[16] V. Chakravarthy, X. Li, R. Zhou, Z. Wu, and M. Temple, "Novel overlay/underlay cognitive radio waveforms using sd-smse framework to enhance spectrum efficiency-part II: analysis in fading channels," *IEEE Transactions on Communications*, vol. 58, no. 6, pp. 1868–1876, 2010.

[17] A. K. Karmokar, S. Senthuran, and A. Anpalagan, "Physical layer-optimal and cross-layer channel access policies for hybrid overlay-underlay cognitive radio networks," *IET Communications*, vol. 8, no. 15, pp. 2666–2675, 2014.

[18] J. Zou, H. Xiong, D. Wang, and C. W. Chen, "Optimal power allocation for hybrid overlay/underlay spectrum sharing in multiband cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1827–1837, 2013.

[19] H. Cho and G. Hwang, "An optimized random channel access policy in cognitive radio networks under packet collision requirement for primary users," *IEEE Transactions on Wireless Communications*, vol. 12, no. 12, pp. 6382–6391, 2013.

[20] S. Xie and Y. Wang, "Construction of tree network with limited delivery latency in homogeneous wireless sensor networks," *Wireless Personal Communications*, vol. 78, no. 1, pp. 231–246, 2014.

[21] J. Shen, H. Tan, J. Wang, J. Wang, and S. Lee, "A novel routing protocol providing good transmission reliability in underwater sensor networks," *Journal of Internet Technology*, vol. 16, no. 1, pp. 171–178, 2015.

[22] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: survey and implications," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

[23] Pratibha, K. H. Li, and K. C. Teh, "Energy-harvesting cognitive radio systems cooperating for spectrum sensing and utilization," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '15)*, San Diego, Calif, USA, December 2015.

[24] L. Mohjazi, M. Dianati, G. K. Karagiannidis, S. Muhaidat, and M. Al-Qutayri, "Rf-powered cognitive radio networks: technical challenges and limitations," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 94–100, 2015.

[25] S. Hu, Y. D. Yao, and Z. Yang, "Cognitive medium access control protocols for secondary users sharing a common channel with time division multiple access primary users," *Wireless Communications and Mobile Computing*, vol. 14, no. 2, pp. 284–296, 2014.

[26] H. A. B. Salameh and M. F. El-Attar, "Cooperative OFDM-based virtual clustering scheme for distributed coordination in cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 8, pp. 3624–3632, 2015.

[27] S. P. Herath and N. Rajatheva, "Analysis of equal gain combining in energy detection for cognitive radio over Nakagami channels," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '08)*, pp. 1–5, New Orleans, La, USA, November 2008.

[28] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless networks with RF energy harvesting: a contemporary survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 757–789, 2015.

[29] S. Wang, Y. Wang, J. P. Coon, and A. Doufexi, "Energy-efficient spectrum sensing and access for cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 2, pp. 906–912, 2012.

[30] S. Park, H. Kim, and D. Hong, "Cognitive radio networks with energy harvesting," *IEEE Transactions on Wireless Communications*, vol. 12, no. 3, pp. 1386–1397, 2013.

[31] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[32] W. B. Chien, C. K. Yang, and Y. H. Huang, "Energy-saving cooperative spectrum sensing processor for cognitive radio system," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 4, pp. 711–723, 2011.

[33] Y. Zou, Y. D. Yao, and B. Zheng, "Cooperative relay techniques for cognitive radio systems: spectrum sensing and secondary user transmissions," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 98–103, 2012.

[34] P. J. Smith, P. A. Dmochowski, H. A. Suraweera, and M. Shafi, "The effects of limited channel knowledge on cognitive radio

system capacity," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 927–933, 2013.

[35] B. Wang, Z. Ji, K. J. R. Liu, and T. C. Clancy, "Primary-prioritized markov approach for dynamic spectrum allocation," in *Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '07)*, pp. 1854–1865, Dublin, Ireland, April 2007.

[36] Y. Song and J. Xie, "ProSpect: a proactive spectrum handoff framework for cognitive radio ad hoc networks without common control channel," *IEEE Transactions on Mobile Computing*, vol. 11, no. 7, pp. 1127–1139, 2012.

[37] M. G. Khoshkholgh, K. Navaie, and H. Yanikomeroglu, "Access strategies for spectrum sharing in fading environment: overlay, underlay, and mixed," *IEEE Transactions on Mobile Computing*, vol. 9, no. 12, pp. 1780–1793, 2010.

[38] Y. Wang, P. Ren, F. Gao, and Z. Su, "A hybrid underlay/overlay transmission mode for cognitive radio networks with statistical quality-of-service provisioning," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1482–1498, 2014.

[39] S. Gmira, A. Kobbane, and E. Sabir, "A new optimal hybrid spectrum access in cognitive radio: overlay-underlay mode," in *Proceedings of the International Conference on Wireless Networks and Mobile Communications (WINCOM '15)*, Marrakech, Morocco, October 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 6575931, 8 pages
http://dx.doi.org/10.1155/2016/6575931

Research Article

Making Image More Energy Efficient for OLED Smart Devices

Deguang Li,[1] Bing Guo,[1] Yan Shen,[2] Junke Li,[1] and Yanhui Huang[1]

[1]*College of Computer Science, Sichuan University, Chengdu 610065, China*
[2]*School of Control Engineering, Chengdu University of Information Technology, Chengdu 610225, China*

Correspondence should be addressed to Bing Guo; guobing@scu.edu.cn

Received 21 September 2016; Revised 11 November 2016; Accepted 22 November 2016

Academic Editor: Laurence T. Yang

Now, more and more mobile smart devices are emerging massively; energy consumption of these devices has become an important consideration due to the limitation of battery capacity. Displays are the dominant energy consuming component of battery-operated devices, giving rise to organic light-emitting diode (OLED) as a new promising display technology, which consumes different power when displaying different content due to their emissive nature. Based on this property, we propose an approach to improve image energy efficiency on OLED displays by perceiving image content. The key idea of our approach is to eliminate undesired details while preserving the region of interest of the image by leveraging the color and spatial information. First, we use edge detection algorithm to extract region of interest (ROI) of an image. Next, we gradually change luminance and saturation of region of noninterest (NON-ROI) of the image. Then we perform detailed experiment and case study to validate our approach; experiment results show that our approach can save 22.5% energy on average while preserving high quality of the image.

1. Introduction

With the rapid development of big data, mobile computing, and internet of things, more and more mobile smart devices are emerging in our life. Smartphones as one of typical smart devices have become very popular in recent years, and more than 1.9 billion of them are being used worldwide today. Gartner predicts that they will grow from 1.9 billion in 2015 to over two billion in 2018 [1]. Most of the smart devices are powered by lithium-ion batteries, while the batteries are limited in size and capacity; thus low energy consumption is an urgent concern for mobile smart devices.

Modern smartphones are equipped with a wide range of I/O components and sensors, such as CPU, displays, Wi-Fi NIC, graphics, Bluetooth, GPS, and audio. Many researchers put forward various strategies to reduce energy consumption of mobile devices, such as Lee and Kim [2] who presented a new approach for energy-efficient real-time HAR on smart mobile devices and Peng et al. [3] who obtained the optimal transmission rate threshold at each detection slot time to reduce network energy consumption of mobile devices. Among all the components of mobile devices, displays are the most power-hungry component, which consume 38–50% of total energy [4, 5]. Unlike liquid crystal display (LCD) panels which require high intensity backlight, the new emerging organic light emitting diode displays (OLED) emit light by their pixels themselves, which do not need an external backlight as the illumination source. Thus this brings us a new opportunity for energy saving, since energy consumption of each pixel of the OLED depends on the content displayed. Each pixel of an OLED display emits three channels of the color: red, green, and blue. Dong and Zhong [6] have pointed that energy consumption of the three colors is different, for example, the black color consumes the lowest energy since the luminance of the three channels is zero, and the white color consumes the highest energy since all the luminance of the three channels is fully filled. Many studies [7–13] show that OLED display technology is widely used in various kinds of smart devices, which will become the mainstream display technology in the future for mobile smart device; thus it is meaningful to explore low energy consumption methods for mobile OLED displays.

Many attempts have proposed to reduce energy consumption of OLED displays from different aspects, mainly through dynamic voltage scaling of OLED displays [7–9, 14, 15], context-aware dimming [10–12, 16–18], and color remapping

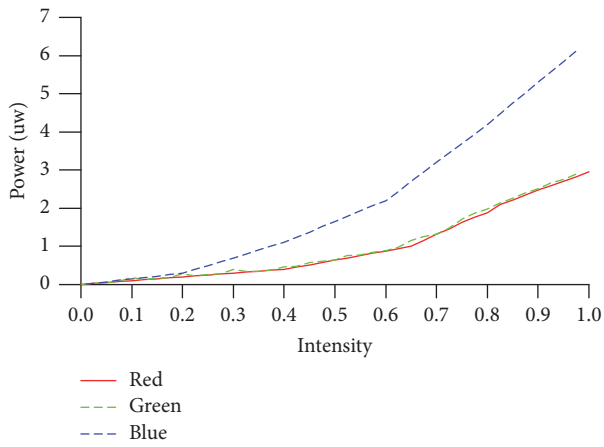


Figure 1: Power consumption for the R, G, and B components of an OLED pixel by different intensity levels.

[13, 15, 19–23]. Among these techniques, dimming is a simple and effective method for energy saving, which reduces luminance of the pixels displayed on OLED, while color remapping is implemented by replacing the energy-saving color schemes with the energy-hungry color schemes. Dong and Zhong [6] first proposed power modeling and optimization for OLED displays, and they noticed that OLED consumed drastically different power when displaying different colors due to their emissive nature. Dimming technique was put forward to reduce energy consumption of OLED displays in [10–12, 16–18], and the essence of dimming technique is to reduce the luminance of three color components (R, G, and B) in each pixel of the OLED, so we could conclude that dimming technique is also color remapping. Compared with color remapping, dimming technique is simple and an effective way, which is used widely in mobile games [11] and videos [22]. Inspired by this, we try to improve image energy efficiency for smart devices by dimming NON-ROI of the image.

In this paper, we present our approach by eliminating undesired details while preserving the region of interest of the image. Our approach consists of two major steps. First, we use edge detection algorithm to extract the ROI of an image; we make use of the classic Canny edge detection algorithm and bilateral filter to get the region of interest of the image. Second, we change the luminance and saturation of NON-ROI of the image gradually. We change the luminance and saturation of the image from the salient region to image boundary so as to ensure maximum image quality, which can reduce the maximum power consumption of NON-ROI area while ensuring that the average structural similarity (MSSIM) of the image is meeting the user's vision requirements. Finally, we perform detailed experiments and case study to validate our approach; experiment results show that our approach can save 22.5% energy on average while preserving high quality of the image, which demonstrates the effectiveness and efficiency of our approach.

The organization of this paper is as follows: related work is presented in Section 2, and our approach is elaborated in Section 3; experiment evaluation and discussion are given in Section 4, and we conclude our research in Section 5.

2. Related Works

2.1. OLED Energy Model. OLED is a new emerging display technology which provides wider viewing angels and better power efficiency than traditional LCD and which is widely used in commercial applications such as displays for various kinds of smart devices and portable digital devices. The main difference between OLED displays and LCD displays is that OLED displays do not require external lighting and the pixels of OLED are emissive by themselves. Each pixel of an OLED consists of three color components, namely, red, green, and blue. Dong and Zhong [6] first proposed power modeling and optimization for OLED displays; they modeled the power contributed by a single pixel, specified in (R, G, and B), shown as formula (1). In order to measure the energy consumption of one image, we also make use of the generic OLED power model proposed by Dong and Zhong.

$$P_{\text{pixel}}(\text{R}, \text{G}, \text{B}) = f(\text{R}) + h(\text{G}) + k(\text{B}) \quad (1)$$

$$P = C + \sum_{i=1}^{n} \{f(\text{R}_i) + h(\text{G}_i) + k(\text{B}_i)\}, \quad (2)$$

where $f(\text{R})$, $h(\text{G})$, and $k(\text{B})$ are power consumption of red, green, and blue devices of the pixel, respectively. And the power consumption of an OLED display with n pixels is (2). C is the static energy consumption which is dominated by a driven current of the control chips and which can be estimated by measuring the energy consumption of a completely black screen. We can get $f(\text{R})$, $h(\text{G})$, and $k(\text{B})$ by measuring the energy consumption for each individual channel with different intensity levels. Figure 1 shows the energy consumption of three colors with different intensity on a μOLED-32028-P1 AMOLED display.

2.2. Energy Saving. Smart devices have become a part of everyone's life, which offer more and more services for our daily life. Particularly for smartphones, there are millions of applications in Google Play and App Store, and these rich applications require battery to provide more energy. The capacity of lithium-ion batteries is still constrained by the size and weight due to smart devices' mobility; thus energy saving of smart devices has become an urgent concern. Many studies have been made to reduce energy consumption of smart devices from different aspects; particularly for OLED, there are mainly three ways for energy saving, namely, dynamic voltage scaling [7–9, 14, 15], color context aware dimming [10–12, 16–18], and color remapping [13, 15, 19–23].

Shin et al. [7] first introduced dynamic driving voltage scaling (DVS) of OLED panel technique; the idea is to scale down the supply voltage and, in turn, dramatically reduce the wasted power caused by the voltage drop across the driver transistor as well as internal parasitic resistance; thus energy is saved on the OLED display panel with only minor changes in the color and luminance of the image; their experiment shows that their method saves up to 52.5% of the OLED

energy while keeping the same image quality for the Lena image. Based on their work, Chen et al. [8] proposed a new fine-grained dynamic voltage scaling method; the key point is partitioning the OLED panel into multiple display areas and adjusting the supply voltage based on the displayed content, and they designed a DVS-friendly OLED driver to enhance the color accuracy of the OLED pixels at the scaled supply voltage. Their experimental results show that, compared to existing global DVS technique, FDVS technique can achieve 25.9%~43.1% more OLED power saving while maintaining a high image quality measured by Structural Similarity Index (SSIM = 0.98). Also in [15], Song and Park presented a decoding model that allows buffering frames to let the CPU run at low frequency to reduce the energy required for video decoding. These two methods focus on the hardware structure of the OLED, which are compatible with our method, since our method focuses on the image. In our study, we also use SSIM to validate the image quality to ensure the effectiveness of our method.

Since power consumption of each pixel of the OLED depends on the color displayed, previous energy saving methods [9–13, 19–22] mainly change the color or luminance of the displayed image. As illustrated in Figure 1, we can clearly observe that blue is the most power-hungry color and energy efficiencies of different colors are different. Dong et al. [13] first took a commercial off-the-shelf QVGA OLED module to adapt GUIs for energy saving; they designed a color-adaptive web browser for mobile OLED displays in [20], and the browser renders web pages with energy-saving color schemes under user-supplied constraints. Similarly, Li et al. [21] proposed an approach for automatically rewriting web applications so as to generate more energy-efficient web pages for mobile smart devices. Wang et al. [19] put forward an approach to reduce power consumption on OLED displays for sequential data visualization by replacing autogenerated color schemes with the most energy-saving color schemes. They first create a multiobjective optimization approach to find the most energy-saving color schemes for given visual perception difference levels and then apply the model in two situations: predesigned color schemes and autogenerated color schemes. These methods are all the specific implementation of color remapping technique, while remapping color is computing intensive and complicate for image processing. Dimming is a simple and effective way, which is widely used for game and video processing in [10–12, 16–18].

Dalton and Ellis [10] first used dimming technique to reduce energy consumption of the displays; they used a web camera to detect the user's face, keeping the laptop's display on as the user is present and turning it off when the user leaves, while this dimming method is coarse-grained. Then fine-grained dimming attempts [11, 12, 16–18] were proposed; Wee and Balan [11] put forward a technique which makes use of saliency, by reducing the brightness of game areas which are not of interest currently to the game player to reduce the power consumption of OLED displays. They assumed that the region of interest was the center of the screen and used a rectangle representing the ROI and then computed a series of dimming boxes (rectangle) from the ROI to the edge of the screen. While our ROI is obtained by edge detection algorithm, which is more accurate than the assumption that ROI is center of the screen, the shape of our dimming boxes is based on the shape of the ROI, which is more helpful to preserve the quality of the image. Betts-LaCroix [12] put forward dimming selected areas of the OLED display according to use's perception, which also results in power savings for OLED display.

Also in [16–18], all authors used dimming technique for power saving. The key idea of [16] is that they also use the notion of saliency to save display power by dimming portions of the applications that are less important to the user, while they used a simple ROI model which assumes that user attention is directed mostly towards the top or bottom portions of the screen when using an application. Choubey et al. [17] noticed that if distance between two point size light sources is less than 0.04–0.05 mm, they will appear as single source for human visual system. Based on human eye's visual acuity, even if some subpixels are turned off, they will not be perceived. Thus a display content and human visual acuity aware technique to reduce OLED panel power consumption was proposed. The crucial step is turning off selective subpixels in specific regions of display. This approach focuses on the manipulation of OLED according to the display content and human visual acuity, while our approach is focused on the image itself. The common point between their method and our method is reducing the power consumption of pixels in specific region of the display. Lin et al. [18] introduced an alternative low-power technique called image pixel scaling, which leverages the flexibility provided by OLED technology to scale down the pixel values of different-shaped regions; this approach is similar to ours. And they proposed the design, algorithm, and implementation of a novel framework called CURA for quality-retaining power saving on mobile OLED displays. Their method is able to display an image without adversely impacting the user's visual experience, while they implemented attention region segmentation, region distortion assessment, and boundary effect elimination to target the ROI of the image, which led to more computational overhead. Compared to their method, our approach is more simple and fast in image processing, while maybe the effect of power saving is not better than theirs.

Most of these studies reduce energy by dimming or turning off the selected areas of the OLED when displaying content, while our approach is different from these methods, since we focus on the image itself and we try to improve image energy efficiency for mobile smart devices initiatively.

3. Our Approach

For a specific image, our approach consisted of two main steps, namely, region of interest (ROI) abstraction and region of noninterest (NON-ROI) dimming. First, we use edge detection algorithm to extract ROI of the image and then adjust the luminance and saturation of NON-ROI of the image smoothly. In our paper, we use classical Canny [24] algorithm for ROI abstraction, dimming NON-ROI of the image by adjusting luminance and saturation (ALS) algorithm; detailed steps are presented below.

3.1. ROI Abstraction. The region of interest of an image is also known as salient region of the image, which shows the main content and the most interesting area of an image. ROI abstraction is the process of preserving the salient region while eliminating undesired details of the image. Usually we use edge detection for feature detection and extraction; edge detection is designed to detect edges or a significant change in discrete regions of a digital image. Typically used edge detection algorithms have Canny edge detection, wavelet transform detection, and fuzzy theory detection and so on. In this paper, we use classical Canny algorithm for image edge detection. Canny operator is considered to be the most classic edge detector, and many of them do comparative analysis always refering to this standard. Canny operator is originally designed for gray image edge detection, with the increasingly wide range of applications of color images, which is also applied to edge detection of color images. In the process of edge detection for color image, we need to convert color image to gray image; main steps of the algorithm are described below.

(1) Image Graying. The process of turning a colorful image into a gray image is called image graying. Detailed steps calculated the average value of the three components (R, G, and B) of each pixel and then assigned the average value to the three components of each pixel. Considering human physiological characteristics, we convert a color image to a gray image according to formula (3). This step is not required for gray images.

$$P_{\text{Gray}} = P_{\text{R}} * 0.299 + P_{\text{G}} * 0.587 + P_{\text{B}} * 0.114. \quad (3)$$

(2) Image Smoothing. Smoothing is blurring the image to remove noise. Canny algorithm uses Gauss filter to smooth the image. In subsequent applications, Tomasi and Manduchi [25] find that Gauss filter obviously blurs the edge and protective effect of high-frequency details is not obvious. In this paper we use bilateral filter for image smoothing, which is a nonlinear, edge-preserving, and noise-reducing for images and is based on the combination of the spatial proximity of the image and the similarity of each pixel, considering the spatial information and the gray similarity. Moreover, it can also achieve the goal of edge-preserving and noise-reducing. The formula of bilateral filter is as follows:

$$I^{\text{filtered}}(x) = \frac{1}{W_p} \sum_{x_i \in \Omega} I(x_i) f_r\left(\left\|I(x_i) - I(x)\right\|\right) g_s\left(\left|x_i - x\right|\right) \quad (4)$$

$$W_p = \sum_{x_i \in \Omega} f_r\left(\left\|I(x_i) - I(x)\right\|\right) g_s\left(\left|x_i - x\right|\right). \quad (5)$$

I^{filtered} is the filtered image and W_p is the normalization factor. I is the original input image to be filtered, x is the coordinates of the current pixel to be filtered, Ω is the window centered in x, f_r is the range kernel for smoothing differences in intensities, and g_s is the spatial kernel for smoothing differences in coordinates.

(3) Nonmaximum Suppression. This step is to find the location of the gray intensity with the sharpest changes in the image and then makes the fuzzy edge become clear through nonmaximum suppression. The gray intensity with sharpest changes is the great changes of the gradient direction of the image; gradient $G(x, y)$ and direction $\theta(x, y)$ of one pixel can be calculated by calculating the gradient and direction in X and Y; the formulas are (6) and (7), where $G_x(x, y)$ and $G_y(x, y)$ are the gradients in the x and y directions, respectively. Then according to the gradient and the angle calculated, take nonmaximum suppression and preserve the local maximum gradient points to realize the edge thinning, which will reduce the edge pixels after these steps and reduce the difficulty of determining the edges.

$$G(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)} \quad (6)$$

$$\theta(x, y) = \arctan\left(\frac{G_y(x, y)}{G_x(x, y)}\right). \quad (7)$$

(4) Double Thresholding. After nonmaximum suppression, edge pixels are quite accurate to present the real edge. However, there are still some edge pixels at this point caused by noise and color variation. In order to get rid of the spurious responses from these bothering factors, it is essential to filter out the edge pixel with the weak gradient value and preserve the edge with the high gradient value. The simplest way to discern between these would be to use a threshold; thus only edges which are stronger than a certain value would be preserved, so the algorithm uses double thresholding to finish this process.

(5) Edge Tracking by Hysteresis. Strong edge pixels are certainly preserved in the final edge image since they are extracted from the true edges in the image. However, for the weak edge pixels, there are still some debates on these pixels which can be extracted from either the true edge or the noise/color variations. In order to get an accurate result, weak edges caused by the noise/color variations should be removed. Generally a weak edge pixel caused by true edges will be connected to a strong edge pixel while noise responses are unconnected. Then blob analysis is applied by looking at a weak edge pixel and its 8-connected neighborhood pixels to track the edge connection.

3.2. NON-ROI Dimming. Adjusting luminance and saturation of the NON-ROI of an image is the key step for energy-saving, in order to ensure the luminance and saturation of the image having a smooth transition. We adjust them from the salient region to the image boundary gradually in order to ensure maximum image quality. In this paper, we take luminance adjustment and saturation adjustment simultaneously, reducing the maximum power consumption of NON-ROI region while ensuring the average structural similarity (MSSIM) of the image to meet user's vision requirements.

(1) Luminance Adjustment. For a given pixel, its luminance is weighted sum of three-color (R, G, and B) pixel values;

the formula is (8), where f is weighted sum function, and for gray image the weighted sum function is (3). From the formula, we know that the greater the value of the three colors, the higher the luminance of the pixel. When we reduce the luminance of one pixel, which directly reduces the value of each color component of the pixel, the luminance adjustment can be expressed as (9).

$$P_{\text{pixel}}(S) = f(\text{R}, \text{G}, \text{B}), \tag{8}$$

$$P'_{\text{pixel}}(S) = f(\text{R}, \text{G}, \text{B}) * \left(1 - X * \left(\frac{n_i}{N}\right)\right). \tag{9}$$

P_{pixel} is the original luminance of one pixel and P'_{pixel} is the luminance after adjusting. X is the adjustment parameter and its range is between 0 and 1. If the value of a is 0, it means we do not adjust, and if the value of a is 1, it means adjust completely and the pixel becomes complete black. N is the number of the adjustment regions of noninterest region of the image, and the reasonable range of N verified by experiment in this paper is 3–15. n_i is the region number of the pixels located, and we assume the value of the first noninterest region adjacent to the domain of interest is 1, and the region number of the following region is increasing in turn, and the last region of the noninterest is N. We continuously increase the value of n_i from the ROI of the image to the image boundary in adjusting process.

(2) Saturation Adjustment. Saturation describes how a pure color is mixed with achromatic components, which is also known as the purity of color. Bright image always has higher purity, and bleak image has lower purity. The goal of adjusting saturation is to reduce the contributions from the power-hungry color components and enhance the power-efficient ones while maintaining overall luminance. From the OLED power model described in Section 2.1, we know that blue is the highest energy consuming color, so we first consider reducing the blue component of one pixel; thus our saturation adjustment method is described as follows:

$$\begin{aligned}
\text{When} \quad & \max\left[P_{\text{in}}(s)(\text{R}, \text{G}, \text{B})\right] = \text{R or G}, \\
& \max\left[P_{\text{in}}(s)(\text{R}, \text{G}, \text{B})\right] - \text{B} > \Delta s; \\
\text{then} \quad & P_{\text{out}}\left(\text{B}'\right) = P_{\text{in}}(\text{B}) - Y * \left(\frac{n_i}{N}\right), \\
& P_{\text{out}}\left(\text{R}' \text{ or } \text{G}'\right) = P_{\text{in}}(\text{R or G}) + Y * \left(\frac{n_i}{N}\right); \\
& P_{\text{out}}\left(\text{R}' \text{ or } \text{G}'\right) = P_{\text{in}}(\text{R or G}) - Y * \left(\frac{n_i}{N}\right).
\end{aligned} \tag{10}$$

Here Δs describes the color difference of three primary colors, in practice which is usually around 60. Y is the adjustment amount, N is number of grading regulation, and n_i is the region number of the pixels located. We also assign the value of the first noninterest region adjacent to the domain of interest to be 1; the region number of the following region is increasing in turn, and the last region of the noninterest is N.

Now we present our dimming algorithm, which consists of the above two adjustment strategies, which is able to adapt to different images since it is continues to change the adjustment parameter according to the MSSIM of the image. Now, we present our algorithm, namely, ALS (adjusting luminance and saturation of NON-ROI of the image). M_{ROI} and $M_{\text{NON-ROI}}$ are the ROI coordinate matrix and the NON-ROI coordinate matrix of the image; $M_i = \{i \mid i = 1, 2, \ldots, N\}$ is the gradual changing matrix set which is obtained by dividing $M_{\text{NON-ROI}}$ into N matrix sets; $Pi(x, y)$ is one pixel in M_i; and we assume the number of each M_i is k. X is the luminance adjustment parameter and Y is the saturation adjustment amount. The pseudocode of ALS algorithm is shown in Algorithm 1.

4. Results and Evaluation

In this section, we first introduce configuration of our experiment. Next, we implement our dimming algorithm to process images for energy saving. In order to compare the energy saving effect of our algorithm, we present the other two adjust methods: AL (adjusting luminance of the whole image) and AS (adjusting saturation of the whole image), and our approach is ALS. Then, we present one example of processing the image "Seagull," listing the energy saved and some important attributes of the image to illustrate our algorithm. Finally, we randomly select 200 images to verify the generality of our method.

4.1. Experiment Settings. First, we need to get the energy consumption model used in our experiments, which is composed of three estimation functions $f(\text{R})$, $h(\text{G})$, and $k(\text{B})$ (in formula (1)). Since these functions depend on specific displays, we measure them on an aμOLED-32028-P1 AMOLED display module with HOIKI 3334 power meter; detailed configuration of our experiment platform is shown in Table 1. Then we calculate the energy consumption by tracking the electrical current values, and we use KA3005P DC power supply to provide stable and controllable voltage, and leverage HOIKI multifunction power measuring instrument to power consumption of the displays. We change the intensity levels from 0 to 1 for testing each color channel in measuring $f(\text{R})$, $h(\text{G})$, and $k(\text{B})$. In each test, we fill the OLED with corresponding color for 60 seconds to calculate the average energy consumption and detailed results are shown in Figure 1. From Figure 1 we know that the power consumption of each color component is a nonlinear; in order to simplify the calculation, we use least squares curve fitting to get linear relation between the color component and its power consumption as shown in Figure 2. Then we implement the three algorithms to validate the effect of energy saving, and we select one image "Seagull" [26] which is selected from the Google gallery (https://image.google.com/) to validate our algorithm.

4.2. Results and Discussion. Figure 3 illustrates the effects of the above three algorithms in processing image Seagull: (a) is the original image, (b) is adjusting luminance of the whole image, (c) is adjusting saturation of the whole image, and

Adjusting luminance and saturation of NON-ROI of the image
Input: original image
Output: energy-efficient image
Algorithm:
(1) $M_{\text{NON-ROI}} \leftarrow$ Get NON-ROI coordinate matrix of the image by ROI Abstraction;
(2) $N \leftarrow$ Calculate the value of grading regulation by dividing the distance from ROI to the image boundary;
(3) $M_i = \{i \mid i = 1, 2, \ldots, N\} \leftarrow$ Get the gradual changing matrix set by dividing $M_{\text{NON-ROI}}$ into N matrix sets;
(4) $X, Y \leftarrow$ Assign initial values to X and Y
(5) **for i** = 1 to N **do**
(6) **for** $Pi(x, y)$ to $Pk(x, y)$ **do**
(7) adjust the luminance of each pixel by Eq. (7);
(8) adjust the saturation of each pixel by Eq. (8);
(9) **end for**
(10) **end for**
(11) **if** check MSSIM meeting use's requirements is true
combine M_{ROI} and $M'_{\text{NON-ROI}}$ of and output image;
(12) **else**
(13) update X and Y, repeat (5), (6), (7), (8), (9), (10);
(14) **end if**

Algorithm 1: Pseudocode of ALS algorithm.

Table 1: Configuration of experiment platform.

OLED	Power meter	Power supply
aμOLED-32028-P1 AMOLED	HOIKI 3334	KA3005P DC
Resolution 320 × 240	Sampling frequency 74.4 kHz	Voltage range 0 V–30 V
Display color 65 K colors	Measurement accuracy ±0.5% rdg	Current range 0 A–5 A
Diagonal size 2.83 inches	Measurement range 1.5000 W–9.000 kW	Setup accuracy
		Voltage: ≤0.5% + 20 mV
		Current: ≤0.5% + 10 mA

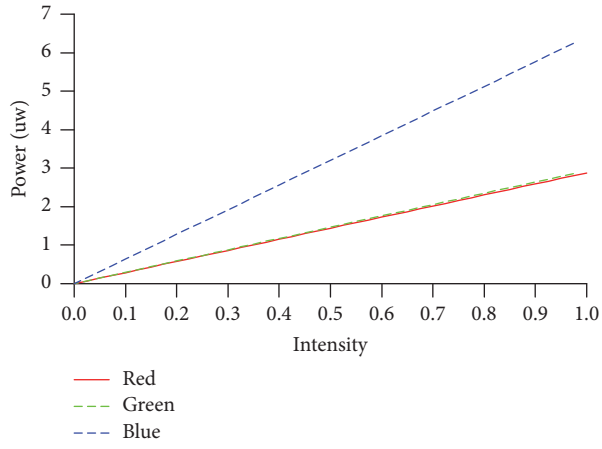
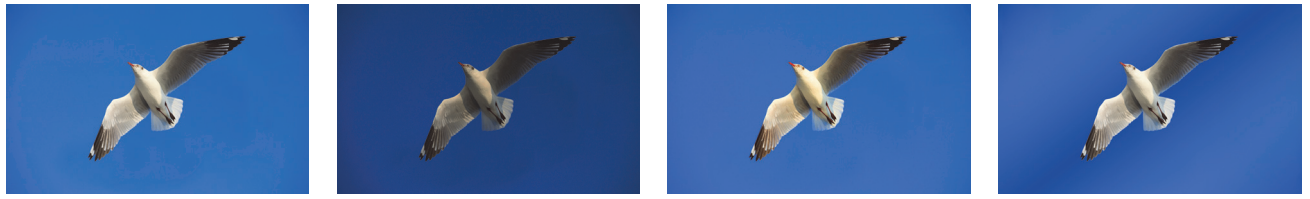


Figure 2: Linear fitted power consumption for the R, G, and B components of an OLED pixel by different intensity levels.

(d) is the image using our approach (ALS). Table 2 shows some important attributes of images (a), (b), (c), and (d). We assume the *SSIM* of original image is 1, and original luminance and saturation of the image are fixed values (we use "—" to stand for the value), and a negative value in the table indicates how much is reduced on the basis of the original value. Table 2 shows experimental results of our method in terms of power consumption and image similarity with the other two methods. First column is the approach name; second column is grading regulation; third column, fourth column, and fifth column are the SSIM of the ROI, NON-ROI, and whole region of the image; sixth and seventh columns are the luminance of the ROI and NON-ROI of the image; eighth and ninth column are the saturation of the ROI and NON-ROI of the image, and the last column is the power consumption of the image.

From Table 2 and Figure 3, we observe that the images processed by the three methods have lower power consumption than original image. The original energy consumption of the image is 2374.836 μw, when the $\text{SSIM}_{\text{OVERVALL}}$ is 0.95 for the image after using three methods; the luminance of image (b) is reduced by 18.76% and energy consumption is reduced by 26.68%; for image (c), the saturation of image is reduced by 21.47% and energy consumption is reduced by 25.96%; and for image (d) the luminance is reduced by 24.24% and saturation is reduced by 19.68%; energy consumption is reduced by 26% when N is 9. At the same time, from Table 2, we observe that when N increases gradually, the MSSIM of the image increases, while power consumption of the image does not change significantly. Thus the brightness and saturation of the image change more smoothly, so the quality



(a) Original (b) AL (c) AS (d) ALS

Figure 3: Images processed by the three methods.

Table 2: Attributes of the images of Figure 3.

Approach	Grading regulation	M_{ROI}	$M_{NON\text{-}ROI}$	$M_{OVERALL}$	L_{ROI}	$L_{NON\text{-}ROI}$	S_{ROI}	$S_{NON\text{-}ROI}$	Power/mw
Original	—	1	1	1	—	—	—	—	2374.836
AL	0	0.95	0.95	0.95	−18.76%	−18.76%	—	—	1741.127
AS	0	0.95	0.95	0.95	—	—	−21.47%	−21.47%	1758.378
ALS	0	0.95	0.95	0.95	−16.58%	−16.58%	−13.42%	−13.42%	1711.341
ALS	3	1	0.86	0.93	—	−19.76%	—	−16.13%	1727.624
ALS	6	1	0.88	0.94	—	−22.16%	—	−18.76%	1746.421
ALS	9	1	0.90	0.95	—	−24.24%	—	−19.68%	1757.378

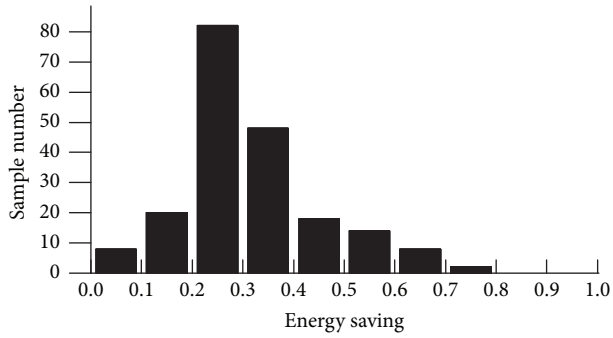


Figure 4: Power reduction ratio distribution of the test samples.

of the image cloud be maintained. Our method preserves the SSIM of ROI completely of the image in dimming process, from Table 2 and Figure 3, which shows that our method is able to reduce the same energy while maintaining high quality of the image.

In order to verify the generality of our method, we performed a case study to assess the effectiveness and users' acceptance of our energy-saving method. We randomly selected 200 images from the Google image database for statistical analysis; for each image, we take the same steps to deal with it. First, we recode energy consumption of the original image and then recode energy reduction of the image processed by our ALS method. Finally, we calculate the energy saving ratio of all the images and the result is presented in Figure 4. Figure 4 is the distribution of the power reduction ratio of 200 images processed by our ALS algorithm; all the MSSIM of the image maintain a value more than or equal to 0.93. From the figure, we can observe that 41% of the experimental samples save energy consumption by 20%–30% and 24% of the samples save energy consumption by 30%–40% and only a few samples have low energy saving since the content of these images is fully filled. The average energy saving of all the test samples is 22.5%, which proves the effectiveness and generality of our method.

Based on the quantitative results of Table 2 and Figure 4, we clearly see that dimming method is effective for energy saving. And our ALS algorithm can be more acceptable than results of uniform dimming methods (AL and AS), since our approach keeps the ROI of the image and adjusts the luminance and saturation gradually, which makes the change of the image more smoothly. In conducting the case study, we find that when white or blue are the main colors of an image, the energy saving effect is obvious, while the energy saving effect is not obvious when the main color of an image is gray. What is more, our algorithm is very effective in dealing with the image with single ROI and the region of ROI is limited; since the region of ROI is limited, we can change much of the region of NON-ROI of the image. Our algorithm is not effective for the image fully filled with content, since the NON-ROI of these kinds of image is very limited. And for the image with no single ROI, we segment an image into a set of ROI regions and for each region applying our algorithm to calculate the maximum change that each region can tolerate according to the average SSIM of the image, which will cause more computing in detecting ROI of the image and calculating the maximum change of NON-ROI of the image.

5. Conclusion

In this work, we propose a new approach to improve image energy efficiency for mobile OLED displays. First we use edge detection algorithm to extract the ROI of an image and then adjust the luminance and saturation of NON-ROI of the image gradually. By this way, we can reduce significant

amounts of energy consumption while preserving high quality of the image. Experiment results show that our approach can save 22.5% energy on average while preserving image quality. Our method is simple and effective for energy saving and can be used for video and other multimedia applications.

Competing Interests

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

Acknowledgments

This work was supported in part by the State Key Program of National Natural Science Foundation of China under Grant no. 61332001, the National Natural Science Foundation of China under Grant nos. 61272104 and 61472050, and the Applied Basic Research Program of Sichuan Province under Grant nos. 2014JY0257, 2015GZ0103, and 2014-HM01-00326-SF.

References

[1] Gartner: more devices will be sold in 2016 than ever—but rifts are appearing, http://www.businessinsider.com/gartner-worldwide-2016-device-forecast-2016-1.

[2] J. Lee and J. Kim, "Energy-efficient real-time human activity recognition on smart mobile devices," *Mobile Information Systems*, vol. 2016, Article ID 2316757, 12 pages, 2016.

[3] Y. Peng, G. Wang, and N. Wang, "Energy-efficient transmission strategy by using optimal stopping approach for mobile networks," *Mobile Information Systems*, vol. 2016, Article ID 8981251, 16 pages, 2016.

[4] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in *Proceedings of the USENIX Conference on USENIX Annual Technical Conference*, vol. 14, p. 21, Boston, Mass, USA, June 2010.

[5] X. Chen, Y. Chen, Z. Ma, and F. C. A. Fernandes, "How is energy consumed in smartphone display applications?" in *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications (ACM HotMobile '13)*, Jekyll Island, Georgia, USA, February 2013.

[6] M. Dong and L. Zhong, "Power modeling and optimization for OLED displays," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1587–1599, 2012.

[7] D. Shin, Y. Kim, N. Chang, and M. Pedram, "Dynamic voltage scaling of OLED displays," in *Proceedings of the Design Automation Conference (DAC '11)*, pp. 53–58, San Diego, Calif, USA, June 2011.

[8] X. Chen, J. Zeng, Y. Chen, W. Zhang, and H. Li, "Fine-grained dynamic voltage scaling on OLED display," in *Proceedings of the 17th Asia and South Pacific Design Automation Conference (ASP-DAC '12)*, pp. 807–812, IEEE, Hong Kong, February 2012.

[9] P. Narra and D. S. Zinger, "An effective LED dimming approach," in *Proceedings of the Industry Applications Conference*, vol. 3, pp. 1671–1676, Seattle, Wash, USA, October 2004.

[10] A. B. Dalton and C. S. Ellis, "Sensing user intention and context for energy management," in *Proceedings of the 9th Conference on Hot Topics in Operating Systems (HOTOS '03)*, pp. 151–156, Lihue, Hawaii, USA, May 2003.

[11] T. K. Wee and R. K. Balan, "Adaptive display power management for OLED displays," in *Proceedings of the 1st ACM SIGCOMM Workshop on Mobile Gaming (MobiGames '12)*, pp. 25–30, Helsinki, Finland, August 2012.

[12] J. Betts-LaCroix, "Selective dimming of oled displays," US Patent 0149223 A1, 2010.

[13] M. Dong, Y.-S. K. Choi, and L. Zhong, "Power-saving color transformation of mobile graphical user interfaces on OLED-based displays," in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED '09)*, pp. 339–342, California, Calif, USA, August 2009.

[14] N. Chang, I. Choi, and H. Shim, "DLS: dynamic backlight luminance scaling of liquid crystal display," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 8, pp. 837–846, 2004.

[15] M. Song and J. Park, "A dynamic programming solution for energy-optimal video playback on mobile devices," *Mobile Information Systems*, vol. 2016, Article ID 1042525, 10 pages, 2016.

[16] T. K. Wee, T. Okoshi, A. Misra, and R. K. Balan, "Focus: A usable & effective approach to OLED display power management," in *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '13)*, pp. 573–582, ACM, Zurich, Switzerland, September 2013.

[17] P. K. Choubey, A. K. Singh, R. B. Bankapur, P. C. Vaisakh, and B. M. Prabhu, "Content aware targeted image manipulation to reduce power consumption in OLED panels," in *Proceedings of the 8th International Conference on Contemporary Computing (IC3 '15)*, pp. 467–471, IEEE, Noida, India, August 2015.

[18] C. H. Lin, C. K. Kang, and P. C. Hsiu, "CURA: a framework for quality-retaining power saving on mobile OLED displays," *ACM Transactions on Embedded Computing Systems*, vol. 15, no. 4, article 76, 2016.

[19] J. Wang, X. Lin, and C. North, "GreenVis: energy-saving color schemes for sequential data visualization on OLED displays," Tech. Rep. TR-12-09, 2012.

[20] M. Dong and L. Zhong, "Chameleon: a color-adaptive web browser for mobile OLED displays," *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, pp. 724–738, 2012.

[21] D. Li, A. H. Tran, and W. G. J. Halfond, "Making web applications more energy efficient for OLED smartphones," in *Proceedings of the ACM 36th International Conference on Software Engineering (ICSE '14)*, pp. 527–538, Hyderabad, India, June 2014.

[22] X. Chen, Y. Chen, and C. J. Xue, "DaTuM: dynamic tone mapping technique for OLED display power saving based on video classification," in *Proceedings of the 52nd ACM/EDAC/IEEE Design Automation Conference (DAC '15)*, pp. 1–6, IEEE, San Francisco, Calif, USA, June 2015.

[23] J. Chuang, D. Weiskopf, and T. Möller, "Energy aware color sets," *Computer Graphics Forum*, vol. 28, no. 2, pp. 203–211, 2009.

[24] L. Ding and A. Goshtasby, "On the canny edge detector," *Pattern Recognition*, vol. 34, no. 3, pp. 721–725, 2001.

[25] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 6th International Conference on Computer Vision*, pp. 839–846, IEEE, Bombay, India, 1998.

[26] Images, https://images.google.com.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 7968108, 12 pages
http://dx.doi.org/10.1155/2016/7968108

Research Article

Speed-Density Model of Interrupted Traffic Flow Based on Coil Data

Chen Yu,[1] Jiajie Zhang,[1] Dezhong Yao,[1] Ruiguo Zhang,[2] and Hai Jin[1]

[1]*Services Computing Technology and System Lab, Big Data Technology and System Lab, Cluster and Grid Computing Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China*
[2]*Siemens Ltd., China Corporate Technology, Wireless Technology and Web of System Wuhan Innovation Center, Wuhan 430074, China*

Correspondence should be addressed to Chen Yu; yuchen@hust.edu.cn

Received 2 September 2016; Accepted 13 November 2016

Academic Editor: Beniamino Di Martino

As a fundamental traffic diagram, the speed-density relationship can provide a solid foundation for traffic flow analysis and efficient traffic management. Because of the change in modern travel modes, the dramatic increase in the number of vehicles and traffic density, and the impact of traffic signals and other factors, vehicles change velocity frequently, which means that a speed-density model based on uninterrupted traffic flow is not suitable for interrupted traffic flow. Based on the coil data of urban roads in Wuhan, China, a new method which can accurately describe the speed-density relation of interrupted traffic flow is proposed for speed fluctuation characteristics. The model of upper and lower bounds of critical values obtained by fitting the data of the coils on urban roads can accurately and intuitively describe the state of urban road traffic, and the physical meaning of each parameter plays an important role in the prediction and analysis of such traffic.

1. Introduction

Flow, speed, and density are known as the basic elements of traffic flow theory. Flow can measure the number of vehicles and the demand for traffic infrastructure. Speed is an important control index in road planning, and it is also an evaluation index of vehicle operation efficiency. Density reflects the intensity of the vehicles on the road and determines traffic management and control measures. The relationships between flow, speed, and density called fundamental diagrams play a very important role in traffic flow theory and traffic engineering. For example, the speed-flow relationship can be used in highway capacity analysis in order to determine the highway service quality, and the speed-density relationship can reflect dynamic change in traffic flow, which can be used to study the disturbance propagation between vehicles. Therefore, sound mathematical models provide a solid foundation for traffic flow analysis and efficient traffic management. The relationship between speed and density which can reflect the quality of service received from the road is attracting considerable research attention.

The earliest speed-density model was a linear model proposed by Greenshields et al. [1] in 1935. The linear model overlaps and classifies the observed data groups, which is proved to be unreasonable, and observation time is a holiday, with a narrow range of representations, so there are some deviations between the derived speed-density relation and the actual situation. Later, the relationship between speed and density was studied in greater depth, and the Greenberg logarithmic model, Edie model, Underwood exponent model, Pipes-Munjal model, modified Greenshields model, Newell model, and so forth, emerged in turn [2, 3]. Heydecker and Addison [4] studied the relationship between speed and density under various speed limits and found that zero speed induces traffic jams, not the other way around. Ma et al. [5] derived a general logistic model of traffic flow characteristics, which includes several traffic flow parameters with clear physical meanings and analyzed the effects of the parameters on speed-density logistic curves. The experimental results showed that this model can well describe the traffic flow characteristics in different states. Shao et al. [6] proposed a speed-density model

under congested traffic conditions combined with the minimum safety spacing constraint, and the experimental results showed that the absolute error of this model was smaller than that of other models fitting the traffic data of two freeways. Wang et al. [7] proposed a family of speed-density models with different numbers of parameters with important physical significance and got good performance in the final experiment.

All of the above studies are based on continuous traffic flow data. These data, also called uninterrupted traffic flow, are traffic flow with no effect of external fixation factors, such as freeway, urban expressway, and so forth. Discontinuous traffic flow, referred to as interrupted traffic flow, is periodically influenced by external fixation factors. The most common interrupted traffic flow is originated by signal lamps of urban intersections. Because of the variety of vehicle types, the periodic effect of signal lamps, shunts in the canal section, and other factors, the characteristics of interrupted traffic flow are very complex compared with uninterrupted traffic flow. In addition, the city is still in a rapid increase in population and, with the development of economy, people are more inclined to self-driving travel, thus more and more vehicles and more and more congestion in the city, which leads to the increase of travel time, the growth of fuel consumption [8], the aggravation of environmental pollution, and other awful issues [9, 10]. Compared with the highway, the urban road has a strong influence on the individual, society, and the environment. Therefore, further study of the characteristics of interrupted traffic flow to provide support for management decisions is particularly important.

Research on interrupted traffic flow has attracted a lot of attention [11–15]. Many scholars see traffic flow located at a certain distance from the intersection as continuous traffic flow, believing that it can be described by continuous traffic flow models. Some of the literature [16, 17] suggests, however, that because of the short distance between intersections in the city and the influence of signal lamps, there are differences between traffic flow located at a certain distance from the intersection and the traffic flow of freeways. Because traffic data are difficult to obtain and for other objective reasons, only a few scholars focus on the speed-density model of discontinuous traffic flow. Wang et al. [18] introduced a four-parameter logit model for complete data fitting and established a speed-density logit model for left-turning, straight, and right-turning traffic flow. However, the experimental data were obtained by VISSIM simulation, and the simulation parameters were not accurate enough to depict the complex city road environment, so the experimental results have certain limitations. Wang et al. [19] thought that the stochastic model would contain more traffic information and put forward the stochastic speed-density model. This stochastic model can generate a probabilistic traffic flow model and can achieve real-time traffic prediction.

In order to provide favorable data analysis and presentation for city traffic, thus to provide decision support for intelligent transportation, characterizing the speed-density relationship of interrupted traffic flow more accurately is full of importance. By analyzing a large amount of data, we propose a description method for a speed-density relationship model which is suitable for discontinuous traffic flow, using the upper and lower curves to describe the upper and lower bounds of velocity values. Because of the discrepant characteristics of the traffic flow in the outer and inner lanes, the coil data of the outer and inner lanes are analyzed and verified.

2. Speed-Density Model

Three basic parameters (flow q, speed u, and density k) are the core content of the traffic flow model. The three have the following relationship:

$$q = k \times u; \tag{1}$$

that is, flow is the product of density and speed. The relationship between two parameters of the three is of great significance in traffic flow, and the relationship between speed and density has received a lot of research attention. Greenshields et al. was an early researcher, who proposed the speed-density linear relationship [1]:

$$u = u_f \times \left(1 - \frac{k}{k_j}\right), \tag{2}$$

where u_f is the speed of free flow, that is, the speed of vehicles unimpeded when the traffic density tends to zero, and k_j is the density of block flow, that is, the density when the traffic flow is blocked and cannot move. As shown in Figure 1, when $k = 0$, the speed can reach the theoretical maximum value, namely, the free flow velocity u_f. The area surrounded by the abscissa, the ordinate of any point on the line, and the coordinate origin is the traffic flow.

Equation (2) can change to

$$k = k_j \times \left(1 - \frac{u}{u_f}\right). \tag{3}$$

Respectively, introduce (2) and (3) into (1), and we get

$$\begin{aligned} q &= u_f \times \left(k - \frac{k^2}{k_j}\right), \\ q &= k_j \times \left(u - \frac{u^2}{u_f}\right). \end{aligned} \tag{4}$$

Equations (4) illustrate that q-k and q-u are quadratic function relations, as shown in Figure 1.

The linear model is too simple, and there are many deficiencies. In order to improve the model, scholars have proposed models based on the linear model but with a higher degree of accuracy. Table 1 lists results for the speed-density model, including the Greenberg model, Underwood model, Northwestern model, Newell's model, Pipes-Munjal model, Drew model, Modified Greenshields model, Del Castillo and Benitez model, Van Aerde model, MacNicholas model. These models with the parameters of important physical meaning provide good results.

Wang et al. [19] established a speed-density logit probability model with four parameters. Wang et al. used VISSIM

Table 1: Speed-density models.

Model	Function	Parameters
Greenshields model (1935)	$u = u_f \times \left(1 - \frac{k}{k_j}\right)$	u_f, k_j
Greenberg model (1959)	$u = u_m \times \ln\left(\frac{k_j}{k}\right)$	u_m, k_j
Underwood model (1961)	$u = u_f \times \exp\left(-\frac{k}{k_m}\right)$	u_f, k_m
Newell's model (1961)	$u = u_f \times \left\{1 - \exp\left[-\frac{\lambda}{u_f} \times \left(\frac{1}{k} - \frac{1}{k_j}\right)\right]\right\}$	u_f, λ, k_j
Northwestern model (1967)	$u = u_f \times \exp\left[-\frac{1}{2} \times \left(\frac{k}{k_0}\right)^2\right]$	u_f, k_0
Pipes-Munjal model (1967)	$u = u_f \times \left[1 - \left(\frac{k}{k_j}\right)^n\right]$	u_f, k_j
Drew model (1968)	$u = u_f \times \left[1 - \left(\frac{k}{k_j}\right)^{n+1/2}\right]$	u_f, k_j
Modified Greenshields model (1995)	$u = u_0 + \left(u_f - u_0\right) \times \left(1 - \frac{k}{k_j}\right)^\alpha$	u_0, u_f, k_j
Del Castillo and Benitez model (1995)	$u = u_f \times \left\{1 - \exp\left[\frac{\lvert C_j \rvert}{u_f} \times \left(1 - \frac{k_j}{k}\right)\right]\right\}$	u_f, C_j, k_j
Van Aerde model (1995)	$k = \frac{1}{c_1 + c_2/\left(u_f - u\right) + c_3 \times u}$	c_1, c_2, c_3, u_f
MacNicholas model (2008)	$u = u_f \times \frac{k_j^n - k^n}{k_j^n + m \times k^n}$	u_f, k_j, n, m

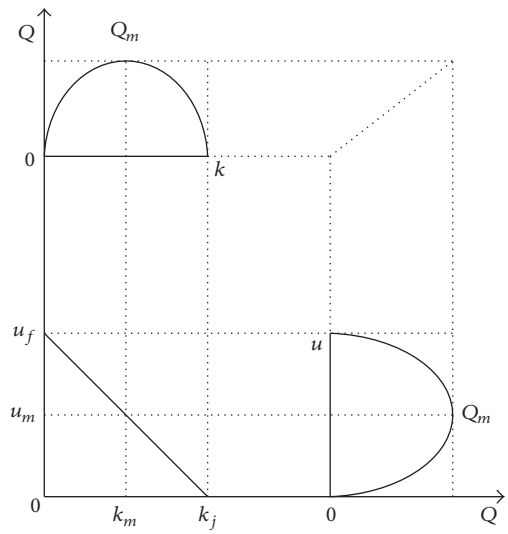


Figure 1: The mapping of speed-density, flow-density, and speed-flow.

simulation software to set up and change six parameters of road traffic, including section length L, stretch section length l, cart rate α, signal period C, the ratio of the time span of left-turn green signal to signal period λ_l, and the ratio of the time span of the straight green signal to signal period λ_s, and established 22 groups of parameters. The simulation results showed that the relationship between speed and density presents an inverse S curve. Therefore, a four-parameter logit model is proposed here to describe the speed-density inverse S curve, and its expression is as follows:

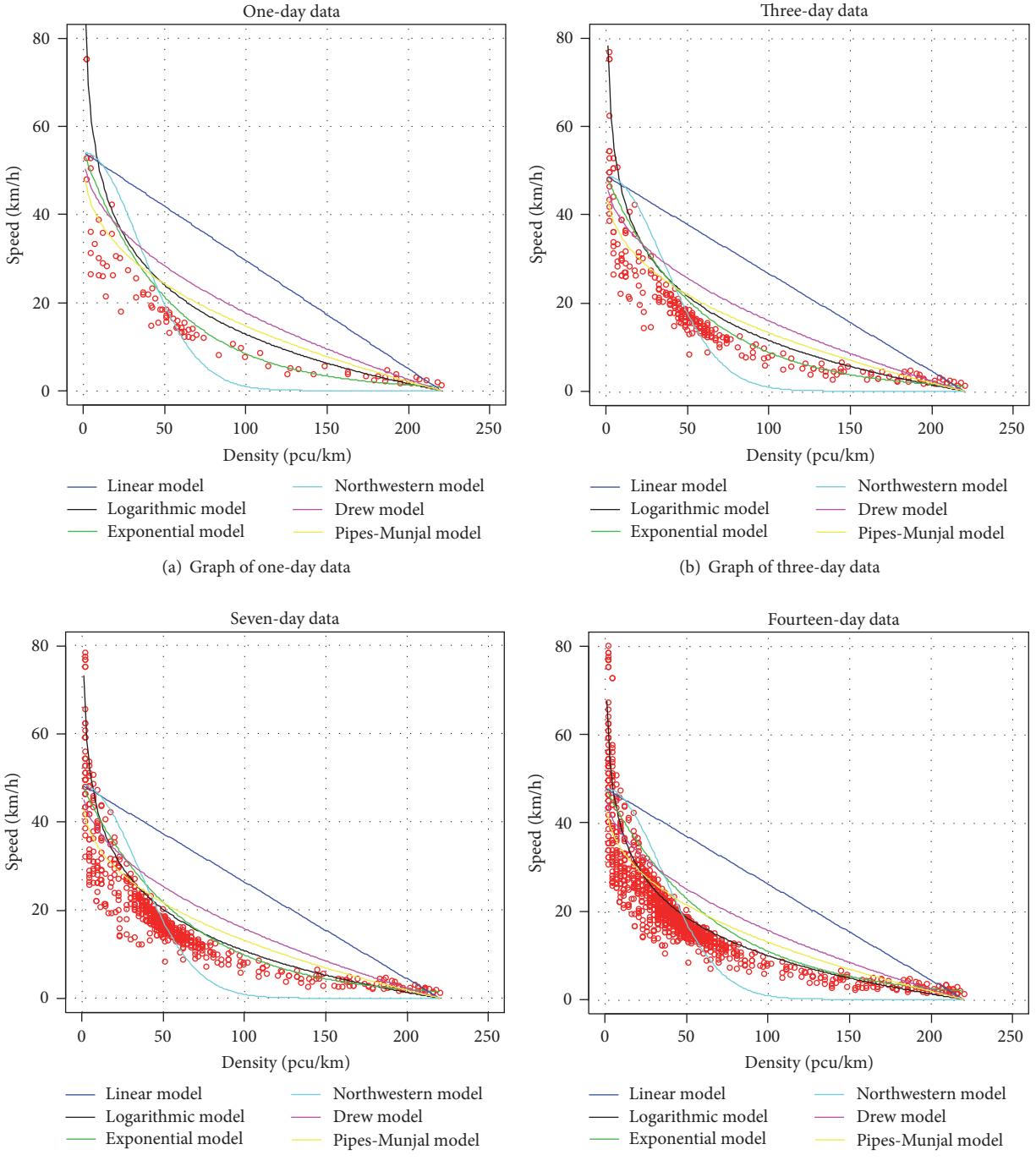
$$u = u_{\min} \times \frac{u_{\max} - u_{\min}}{1 + \exp\left(\left(N - N_w\right)/\theta\right)}, \tag{5}$$

where $u_{\min}$ is the mean value of the minimum speed, $u_{\max}$ is the mean value of the maximum speed, N is the flow value of a section, N_w is the flow value at the inflection point of the curve, and θ is a parameter determining curve shape.

Then, the data obtained from the 22 groups of simulation parameters were fitted. The four parameters ($u_{\min}$, $u_{\max}$, N_w, and θ) were calculated for each simulation environment. N_w and θ were, respectively, fitted in left-turn, straight, and right-turn cases, and the fitting results are as follows:

$$N_w = \begin{cases} 0.7146, & \text{left-turning} \\ -0.2231\alpha + 0.1989, & \text{straight} \\ -0.00011L + 0.0078l - 0.2113\alpha + 0.2282, & \text{right-turning,} \end{cases} \tag{6}$$

$$\theta = \begin{cases} 0.0664, & \text{left-turning} \\ -0.000032L + 0.085l, & \text{straight} \\ 0.045l, & \text{right-turning.} \end{cases}$$



(a) Graph of one-day data

(b) Graph of three-day data

(c) Graph of seven-day data

(d) Graph of fourteen-day data

Figure 2: Comparison of six speed-density models.

3. The Description Method of the Speed-Density Model for Interrupted Traffic Flow

3.1. The Characteristics of the Data of Interrupted Traffic. Coil data for one day, three days, seven days, and fourteen days were selected to compare and analyze the discontinuous flow data and the existing six speed-density models, as shown in Figure 2. We found the following:

(1) The six models' performance was poor when the coil data of interrupted traffic flow were fitted, illustrating that although suitable for uninterrupted traffic flow

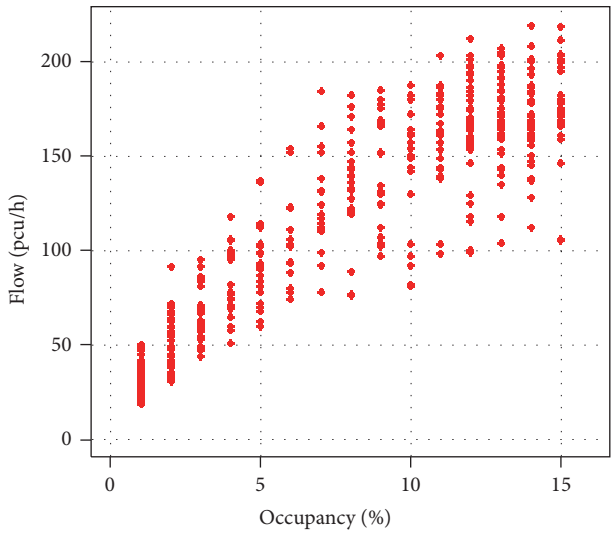


Figure 3: Flow-occupancy graph of small density.

they are unsuitable for describing the speed-density relation of interrupted traffic flow because of diverse data sources, different traffic environments, or other factors. In contrast, the logarithmic model gave the best performance and the linear model gave the worst performance.

(2) The interval value of critical densities k_m of one-day, three-day, seven-day, and fourteen-day data sets was [62.56 pcu/km, 71.23 pcu/km], and most of the data were located in $k < k_m$ range, meaning unimpeded flow data accounted for the absolute proportion, so the traffic flow of the location coil was in a state of flow most of the time.

(3) When $k < k_m$, with the increase of density, the velocity decreased sharply; when $k > k_m$, as the density increased, the velocity decreased slowly, and the speed variation amplitude was very small.

(4) When the density was small, the speed had a large range of values, of which the largest was [23 km/h, 72 km/h]. We filtered out the small-density data to obtain a scatter diagram of flow and occupancy which were directly collected by a loop detector, as shown in Figure 3. In Figure 3 it is obvious that the loop detector acquires large-range flow values for the same occupancy value, and the largest range can reach 100 pcu/h. Hence, after calculating speed and density by density formula and velocity formula, speed accordingly has a large range of values for the same density in speed-density diagram.

(5) In addition, the density values were found to be near a number of points, and the difference between adjacent points was approximately equal to a certain value.

From the above analysis, we found that, because of the big differences between uninterrupted and interrupted traffic flow, existing models suitable for uninterrupted traffic flow are unsuited for describing the speed-density relation of interrupted traffic flow. What is more, the flow collected by a loop detector has a large range of values. Therefore, for the speed-density relation of interrupted traffic flow, we must find a new descriptive method.

3.2. Description Method of Speed-Density Relationship for Interrupted Traffic Flow. Because of the difference between the uninterrupted and interrupted traffic flow and the volatility of speed, the speed-density relationship cannot be adequately described by a single model, so we use two curves, u^{upper} and u^{lower}, to describe the supremum and infimum of velocity values:

$$\begin{aligned} u^{\text{upper}} &= g^{\text{upper}}\left(U^{\text{upper}}\right), \\ u^{\text{lower}} &= g^{\text{lower}}\left(U^{\text{lower}}\right), \end{aligned} \tag{7}$$

where U^{upper} and U^{lower} are, respectively, the upper and lower bounds of velocity and g^{upper} and g^{lower} are fitting functions.

Divide the density interval $[k_{\min}, k_{\max}]$ into n connected intervals $k_1, k_2, \ldots, k_n$. Partition data D as $D_1, D_2, \ldots, D_n$ by density intervals, and correspondingly get speed sets $U_1, U_2, \ldots, U_n$, causing that, for any $i \in (1, 2, \ldots, n)$, we have

$$W\left(U_i\right) > W_\alpha, \tag{8}$$

where $W(U_i)$ is used test for U_i with the Shapiro-Wilk normal test method. Sort m independent observations in U_i by nondescending order, recorded as $x_1, x_2, \ldots, x_m$, and construct the W-test statistic

$$W = \frac{\left[\sum_{i=1}^{m} a_i \times \left(x_{m+1-i} - x_i\right)\right]^2}{\sum_{i=1}^{m} a_i \times \left(x_i - \overline{x}\right)^2}, \tag{9}$$

where a_i is the coefficient when sample size is m. When the population distribution is normal distribution, the value of W should be close to one. α quantile W_α of statistic W can be obtained by the look-up table method. When $W \le W_\alpha$, the original hypothesis should be rejected at the significant level, indicating that U_i does not obey normal distribution; when $W > W_\alpha$, the original hypothesis cannot be rejected, and U_i satisfies normal distribution.

Under the conditions of (8), for every $i \in (1, 2, \ldots, n)$, extract the *upper* quantile u_i^{upper} and *lower* quantile u_i^{lower} as the upper and lower critical values of speed for density interval k_i.

$$\begin{aligned} u_i^{\text{upper}} &= q\text{norm}\left(\text{upper}, \text{mean}\left(U_i\right), \text{sd}\left(U_i\right)\right), \\ u_i^{\text{lower}} &= q\text{norm}\left(\text{lower}, \text{mean}\left(U_i\right), \text{sd}\left(U_i\right)\right), \end{aligned} \tag{10}$$

where $q$norm( ) is quantile function, mean( ) calculates the mean value of U_i, and sd( ) calculates the variance of U_i.

Get the upper bound and lower bound sets

$$\begin{aligned} U^{\text{upper}} &= \left\{u_i^{\text{upper}},\ i = 1, 2, \ldots, n\right\}, \\ U^{\text{lower}} &= \left\{u_i^{\text{lower}},\ i = 1, 2, \ldots, n\right\}. \end{aligned} \tag{11}$$

Fit U^{upper} and U^{lower} using the nonlinear least square method. The tabulated function $u_i = u(k_i)$, $i = 1, 2, \ldots, n$, is

available by (10). Then we need to obtain the fitting function, $g(k) = a_0 + a_1 \times g_1(k) + \cdots + a_p \times g_p(k)$, making the sum of squared deviations

$$S\left(a_0, a_1, \ldots, a_p\right) = \sum_{i=1}^{n} \left[g\left(k_i\right) - u_i\right]^2 = \sum_{i=1}^{n} \left[a_0 + a_1 \times g_1\left(k_i\right) + \cdots + a_p \times g_p\left(k_i\right) - u_i\right]^2. \tag{12}$$

Take the minimum, of which $g_1(k), g_2(k), \ldots, g_p(k)$ are p nonmergeable monomials of variable k, and $a_0, a_1, \ldots, a_p$ are the coefficients of monomials. S is a nonnegative polynomial of $a_0, a_1, \ldots, a_p$, so there must be a minimum value. Respectively, calculate partial derivatives of S for $a_0, a_1, \ldots, a_p$, and make them equal to zero.

$$\frac{\partial S}{\partial a_i} = 0, \quad i = 0, 1, \ldots, p. \tag{13}$$

Equation (13) is expanded as follows:

$$\begin{aligned}
&\frac{\partial S}{\partial a_0} = 2 \times \sum_{i=1}^{n} \left[a_0 + a_1 \times g_1\left(k_i\right) + \cdots + a_p \times g_p\left(k_i\right) - u_i\right] = 0,\\
&\frac{\partial S}{\partial a_1} = 2 \times \sum_{i=1}^{n} \left[a_0 + a_1 \times g_1\left(k_i\right) + \cdots + a_p \times g_p\left(k_i\right) - u_i\right] \times g_1\left(k_i\right) = 0,\\
&\qquad \vdots\\
&\frac{\partial S}{\partial a_p} = 2 \times \sum_{i=1}^{n} \left[a_0 + a_1 \times g_1\left(k_i\right) + \cdots + a_p \times g_p\left(k_i\right) - u_i\right] \times g_p\left(k_i\right) = 0.
\end{aligned} \tag{14}$$

Continue to expand (14):

$$\begin{aligned}
&a_0 \times n + a_1 \times \sum_{i=1}^{n} g_1\left(k_i\right) + \cdots + a_p \times \sum_{i=1}^{n} g_p\left(k_i\right) = \sum_{i=1}^{n} u_i,\\
&a_0 \times \sum_{i=1}^{n} g_1\left(k_i\right) + a_1 \times \sum_{i=1}^{n} \left[g_1\left(k_i\right)\right]^2 + \cdots + a_p \times \sum_{i=1}^{n} \left[g_p\left(k_i\right) \times g_1\left(k_i\right)\right] = \sum_{i=1}^{n} \left[u_i \times g_1\left(k_i\right)\right],\\
&\qquad \vdots\\
&a_0 \times \sum_{i=1}^{n} g_p\left(k_i\right) + a_1 \times \sum_{i=1}^{n} \left(g_1\left(k_i\right) \times g_p\left(k_i\right)\right) + \cdots + a_p \times \sum_{i=1}^{n} \left[g_p\left(k_i\right)\right]^2 = \sum_{i=1}^{n} \left[u_i \times g_p\left(k_i\right)\right]
\end{aligned} \tag{15}$$

and get its matrix form

$$\begin{bmatrix}
n & \sum_{i=1}^{n} g_1\left(k_i\right) & \cdots & \sum_{i=1}^{n} g_p\left(k_i\right)\\
\sum_{i=1}^{n} g_1\left(k_i\right) & \sum_{i=1}^{n} \left[g_1\left(k_i\right)\right]^2 & \cdots & \sum_{i=1}^{n} \left[g_p\left(k_i\right) \times g_1\left(k_i\right)\right]\\
\vdots & \vdots & \vdots & \vdots\\
\sum_{i=1}^{n} g_p\left(k_i\right) & \sum_{i=1}^{n} \left[g_1\left(k_i\right) \times g_p\left(k_i\right)\right] & \cdots & \sum_{i=1}^{n} \left[g_p\left(k_i\right)\right]^2
\end{bmatrix} \times \begin{bmatrix} a_0\\ a_1\\ \vdots\\ a_p \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} u_i\\ \sum_{i=1}^{n} \left[u_i \times g_1\left(k_i\right)\right]\\ \vdots\\ \sum_{i=1}^{n} \left[u_i \times g_p\left(k_i\right)\right] \end{bmatrix}. \tag{16}$$

Solve (16), and $a_0, a_1, \ldots, a_p$ are available.

Using the above least square method to fit U^{upper} and U^{lower}, respectively, obtains

$$\begin{aligned}
u^{\text{upper}}(k) &= g^{\text{upper}}\left(k, a_0, a_1, \ldots, a_p\right),\\
u^{\text{lower}}(k) &= g^{\text{lower}}\left(k, a_0, a_1, \ldots, a_p\right)
\end{aligned} \tag{17}$$

and upper and lower curves $u^{\text{upper}}(k)$ and $u^{\text{lower}}(k)$, which is the speed-density relation description model.

4. Experiment and Analysis

The experiment data is collected by the coil detectors underground closed to Optical Valley Walking Street in Wuhan, China. Coil detectors collect data every 15 minutes, recording *time*, *flow*, *occupancy*, and so forth, as shown in Table 2.

Use the method in [20, 21] to calculate speed and density, and the ratio of the amount of data between two model curves to the total amount of experiment data is used to describe the performance of model. The loop detector in the outer lane measures the traffic flow of straight and right-turning lanes, and the loop detector in the inner lane measures the traffic flow of the left-turning lane. The traffic flow characteristics of two loop detectors must have certain differences. Therefore, analyze the coil data of both the outer lane and the inner lane to find the diversity of their speed-density relationship.

4.1. Coil Data Analysis of the Outer Lane. The experimental steps are as follows.

Step 1. Analyze coil data of the outer lane and find that density values are clustered at a number of points $k_1, k_2, \ldots, k_n$,

Table 2: The data example.

Date	Week	Flow	Occupancy	Minute	id	Hour
2014/11/20	4	132	7	0:00:00	41751051	0
2014/11/20	4	91	5	0:15:00	41751051	0
2014/11/20	4	98	7	0:30:01	41751051	0
2014/11/20	4	103	5	0:45:01	41751051	0
2014/11/20	4	77	6	1:00:00	41751051	1
2014/11/20	4	71	4	1:15:00	41751051	1
2014/11/20	4	64	3	1:30:01	41751051	1
2014/11/20	4	40	75	1:45:01	41751051	1

where the mean value of the difference between the adjacent points is about 2.5 pcu/km. Divide density k into a number of intervals with length 2.5 pcu/km by $k_1, k_2, \ldots, k_n$.

Step 2. Correspondingly split data D into small data sets $D_1, D_2, \ldots, D_n$ according to density segmentations, and get data sets of speed $U_1, U_2, \ldots, U_n$.

Step 3. Execute a distribution test for U_i where the result shows that one data set is too small to meet the requirements of the test. Merge the adjacent density segments in Step 2 to enlarge the amount of the small data set. Redo the distribution test for the new data set, more than 80% of which meets the normal distribution, with totally 95% of the total data satisfying the normal distribution, which makes it reasonable to consider all the small data set satisfying the normal distribution.

Step 4. Get two quantiles u_i^{upper} and u_i^{lower} of speed set U_i as *upper* and *lower* critical values of velocity for density k_i.

Step 5. Then have upper and lower critical value set $U^{\text{upper}} = \sum_{i=1}^{n} u_i^{\text{upper}}$ and $U^{\text{lower}} = \sum_{i=1}^{n} u_i^{\text{lower}}$.

Step 6 (fit u^{upper} and u^{lower}). Because the loop detector is located near commercial street which has heavy traffic, we use the logarithmic model to formulize the data.

$$u^{0.95} = 14.204 \times \ln\left(\frac{216.412}{k}\right),$$
$$u^{0.05} = 7.169 \times \ln\left(\frac{254.497}{k}\right). \tag{18}$$

Figure 4 shows the validation result of the speed-density logarithmic model of the outer lane when upper value = 0.95 and lower value = 0.05. Equations (18) correspondingly are the green and blue curves in Figure 4, which is the speed-density model of interrupted traffic flow created by the new description method. Significant test results indicate that P values of two regression coefficients of two curves are minima ($P < 2e - 16$), which means that coefficients are significant and two log models constructed with density as the independent variable are applied to estimate velocity as the dependent variable.

The coil data of the outer lane for two weeks, four weeks, six weeks, and eight weeks are, respectively, selected

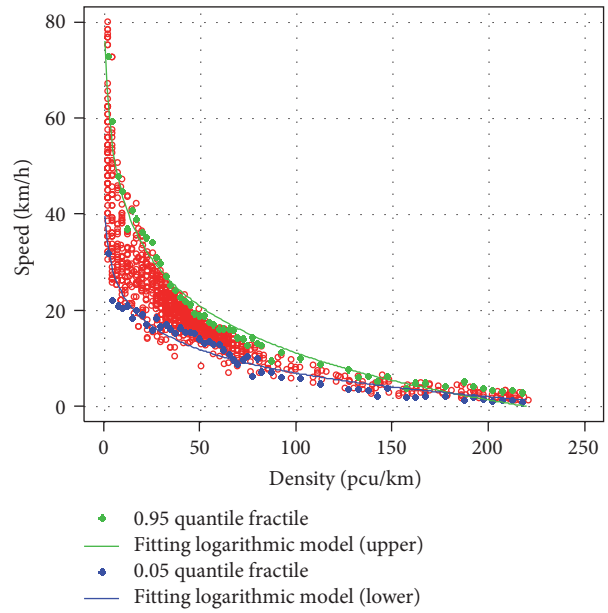


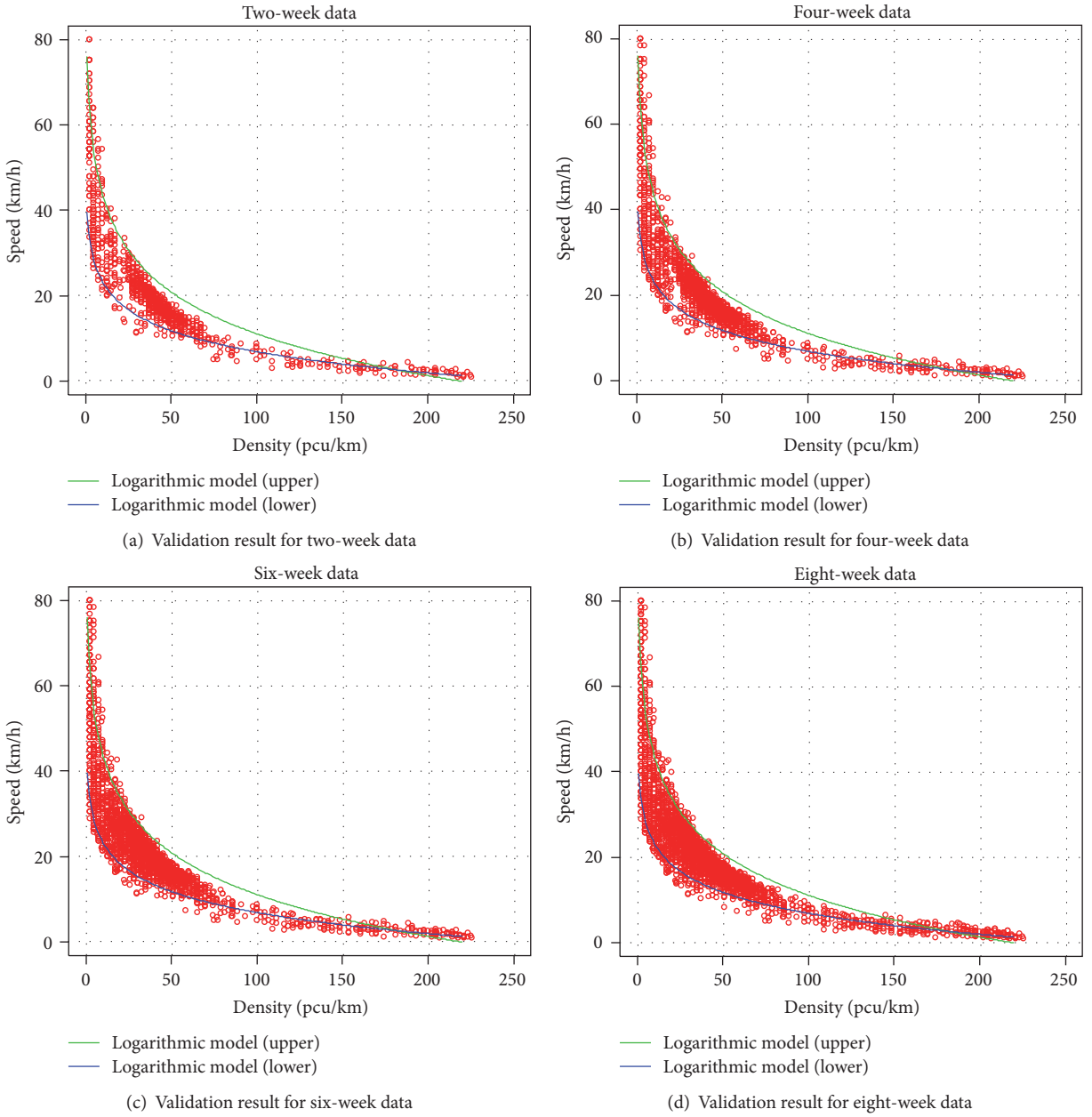
Figure 4: Speed-density logarithmic model of the outside lane.

and four groups of parameters are established for model validation. Table 3 gives the ratio of the data between two logarithmic curves to the total amount of data in each case. Make a longitudinal observation; it is obvious that, with *upper* value increasing and *lower* value decreasing, the proportion increases accordingly, where amplitudes are obvious, respectively, 7.2%, 6.2%, and 6.9%. On the other hand, the main transverse trend is that the proportion increases along with the increase of experiment data loosely, where, however, *six-week data* has the best performance. The above suggests that the two logarithmic models are able to describe the speed-density relation of the outer lane. Figure 5 shows the four groups' validation results when upper value = 0.95 and lower value = 0.05.

4.2. Coil Data Analysis of the Inner Lane. We select coil data of the inner lane and follow Steps 1 to 5 as for the outer lane. When fitting sets u^{upper} and u^{lower} at Step 6, we find that the speed-density models proposed by scholars all have poor performance with goodness of fit of less than 0.5,

Table 3: Validation results of the model of the outer lane.

Parameters	Two-week data	Four-week data	Six-week data	Eight-week data	Average
Upper value = 0.80 Lower value = 0.20	61.9%	65.8%	66.2%	66.1%	65.0%
Upper value = 0.85 Lower value = 0.15	68.9%	73.1%	73.6%	73.3%	72.2%
Upper value = 0.90 Lower value = 0.10	74.5%	78.8%	80.4%	79.7%	78.4%
Upper value = 0.95 Lower value = 0.05	84.2%	85.1%	86.7%	85.3%	85.3%



(a) Validation result for two-week data

(b) Validation result for four-week data

(c) Validation result for six-week data

(d) Validation result for eight-week data

Figure 5: Validation result of the speed-density logarithmic model of the outside lane.

Table 4: Validation results of the model of the inner lane.

Parameters	Two-week data	Four-week data	Six-week data	Eight-week data	Average
Upper value = 0.80 Lower value = 0.20	80.0%	77.9%	81.3%	78.8%	79.5%
Upper value = 0.85 Lower value = 0.15	82.0%	82.3%	85.2%	83.1%	83.2%
Upper value = 0.90 Lower value = 0.10	83.8%	84.9%	86.4%	85.3%	85.1%
Upper value = 0.95 Lower value = 0.05	89.0%	90.3%	88.9%	89.8%	89.5%

which suggests that a single model cannot accurately describe the quantile set of the coil data. Thus we consider using a segmentation model.

In the density-flow curve there is a critical density k_m, which is the density of maximum traffic flow, as shown in Figure 1. When the density $k < k_m$, the traffic is in a state of flow; when $k > k_m$, the traffic flow gradually becomes crowded. Therefore, consider using k_m as the critical value of the subsection.

A density-flow curve is obtained by local polynomial regression fitting, and the density value at the curve vertex is just k_m. Take k_m as the critical value and piecewise analyze $u^{\text{upper}}_{k<k_m}$, $u^{\text{lower}}_{k<k_m}$, $u^{\text{upper}}_{k>k_m}$, and $u^{\text{lower}}_{k>k_m}$. The analysis shows that the quantile set $u^{\text{upper}}_{k<k_m}$ and $u^{\text{lower}}_{k<k_m}$ agrees with the exponential model, and the quantile set $u^{\text{upper}}_{k>k_m}$ and $u^{\text{lower}}_{k>k_m}$ has good agreement with the logarithmic model.

$$u^{0.95} = \begin{cases} 69.647 \times e^{-k/14.449} + 12.716, & k < k_m \\ 8.227 \times \ln\left(\dfrac{254.971}{k}\right), & k \geq k_m, \end{cases}$$
$$u^{0.05} = \begin{cases} 51.08 \times e^{-k/100.10} - 22.45, & k < k_m \\ 5.337 \times \ln\left(\dfrac{243.306}{k}\right), & k \geq k_m. \end{cases} \quad (19)$$

Figure 6 shows the fitting result of a segmentation model of the outer lane when upper value = 0.95 and lower value = 0.05, and (19) are the models corresponding to the green curve and blue curve in Figure 6, which is the speed-density model of interrupted traffic flow via the new description method. P value of each parameter is very small, suggesting the coefficient is very significant.

The coil data of the inner lane for two weeks, four weeks, six weeks, and eight weeks are, respectively, selected and four groups of parameters are established for the model validation, the same as that for the outer lane. Table 4 gives the ratio of the data between two logarithmic curves to the total amount of data in each case. Comparing the result with that of the outer lane, we find that the validation results of the model of the inner lane are better with greater ratio.

Take a longitudinal observation; similarly, it is obvious that with *upper* value increasing and *lower* value decreasing, the proportion increases accordingly, where amplitudes are smaller than that of outer lane, respectively 3.7%, 1.9%, and 4.3%. The main transverse trend is the same as outer lane

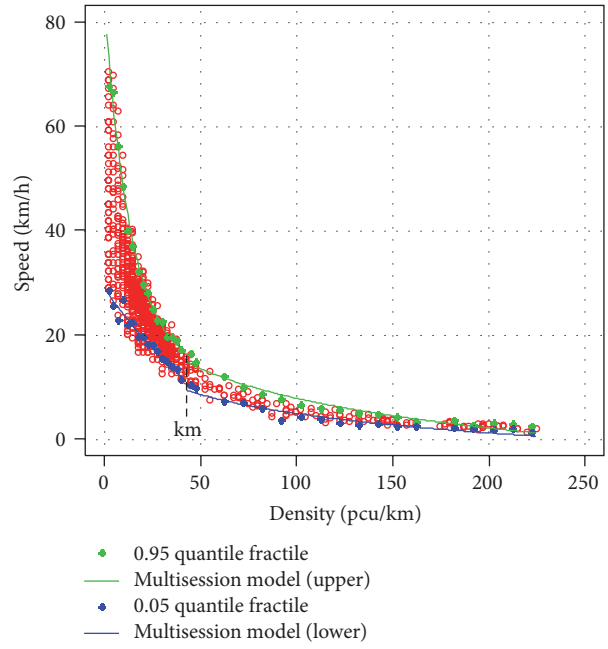


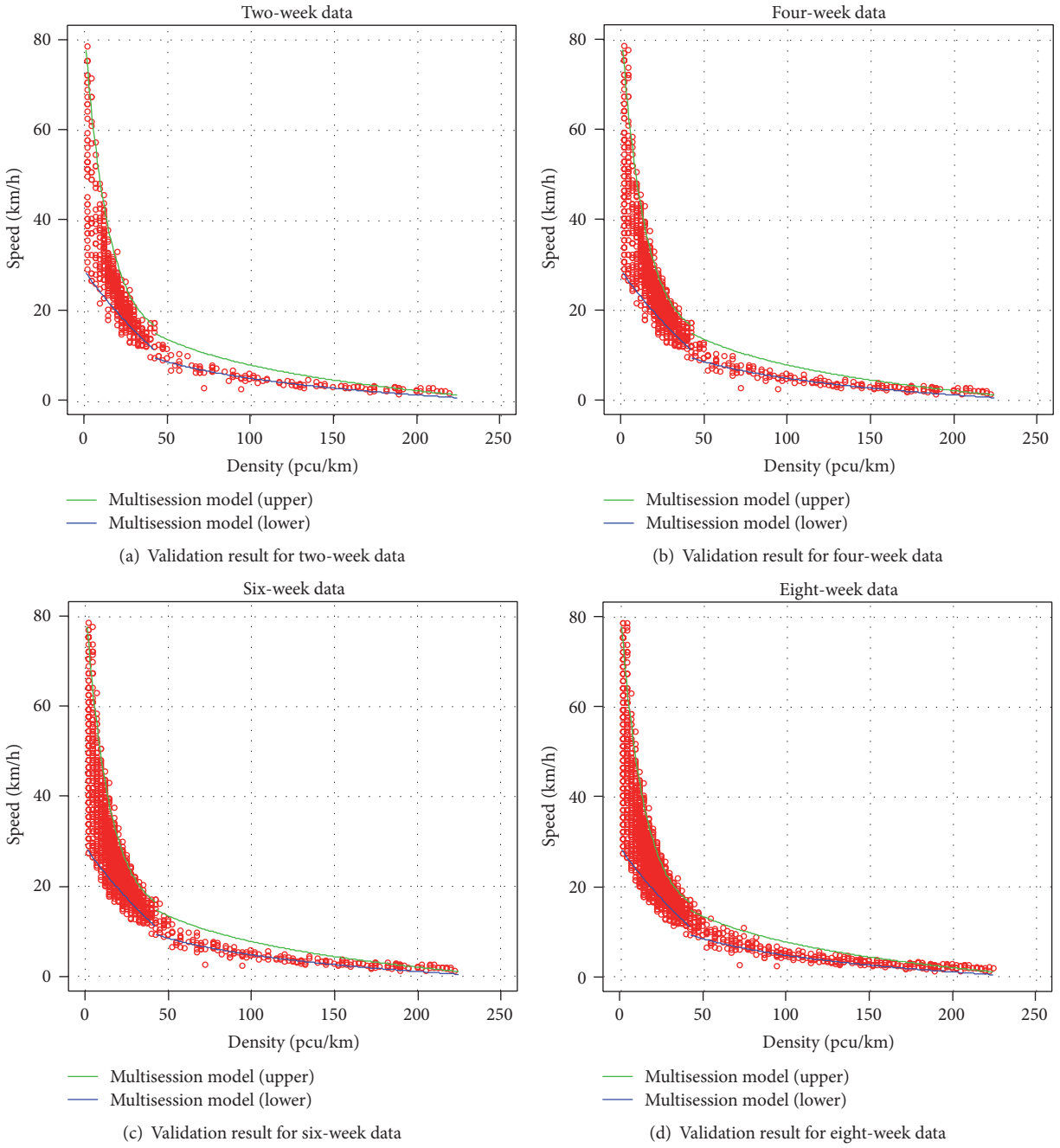
Figure 6: Speed-density multisession model of the inside lane.

except the case of upper value = 0.95 and lower value = 0.05. The result indicates that the two segmentation models are suitable for describing the speed-density relation of the inner lane. Figure 7 shows the four groups' validation results when upper value = 0.95 and lower value = 0.05.

4.3. Experimental Result Analysis

4.3.1. Difference between the Models of the Outer Lane and the Inner Lane. The loop detector of the outer lane measures right-turning and straight lanes, and the coil is located in a road adjacent to a commercial pedestrian street with a heavy flow of people and traffic. A logarithmic model is applied to describe traffic flow with large density, and therefore it is accepted that the coil data of the outer lane satisfy the logarithmic model.

The loop detector of the inner lane measures the left-turning lane which also has heavy traffic flow. The speed-density relation of the inner lane does not satisfy the single log model but is suitable for the segmentation model. The



(a) Validation result for two-week data

(b) Validation result for four-week data

(c) Validation result for six-week data

(d) Validation result for eight-week data

Figure 7: The validation result of the speed-density multisession model of the inside lane.

exponential models of (19) are two-item types with intercepts, instead of Underwood's monomial exponential model. They suggest that the traffic flow of the inner lane differs from the traffic flow of the freeway and outer lane.

4.3.2. Characteristic Analysis of Traffic Flow. The critical density k_m of the outer lane in Figure 6 is 53.6 pcu/km, and most of the density values are less than k_m or in a small range of k_m; similarly, for the inner lane, most of the density values are smaller than k_m, and the data in the range $k > k_m$ are sparse. This illustrates that (1) most of the time the road segment where loop detectors located is unblocked, where the data with big density values which may lead to congestion is just a small proportion, and, (2) compared with the outer lane, the proportion of the density $k < k_m$ of the inner lane is greater, illustrating that the inner lane is more unimpeded than the outer lane.

In summary, the new description method can satisfactorily describe the speed-density relation of interrupted traffic, where the speed-density relation of the outer lane meets the

logarithmic model and the inner lane meets the segment model. What is more, the road segment where loop detectors located is unblocked at most of the time; the inner lane is more unimpeded than the outer lane.

5. Conclusion

In this paper, the characteristics of urban interrupted flow data were analyzed, and it was found that they differ from the data of uninterrupted flow. Since the existing classical models cannot describe them very well, a description method of speed-density relation for interrupted traffic flow was proposed where the upper and lower curves were used as the upper and lower bounds of the predicted speed. In this method, the speed was divided into small data sets which satisfied the normal distribution, and two quantiles of normal distribution were obtained as the predicted values. Then two quantile sets were fitted to get two curves as the speed-density relation model of the interrupted traffic flow. Finally, the coil data of the outer and inner lanes were applied for model validation. The results showed that the new method can give a good description of the speed-density relationship of interrupted traffic flow and get different model results for the outer lane and inner lane, whereby the speed-density relation of the outer lane satisfies the logarithmic model and the inner lane satisfies the segment model instead of the single model, where when the density is less than critical density, it conforms to the exponential model and otherwise the logarithmic model. The fitting results of the internal and external lanes were analyzed in combination with the actual local road environment and traffic flow theory. So this model can provide favorable data analysis and presentation for city traffic, thus to provide decision support for intelligent transportation.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

The work is partly supported by NSFC (no. 61472149), the Fundamental Research Funds for the Central Universities (2015QN67), the Wuhan Youth Science and Technology Plan (2016070204010132), and the National 863 Hi-Tech Research and Development Program under Grant 2015AA01A203.

References

[1] D. B. Greenshields, R. J. Biddins, S. W. Channing et al., "A study in highway capacity," *Highway Research Board Proceedings*, vol. 14, no. 1, pp. 448–477, 1935.

[2] H. Z. Wang, D. h. Ni, Q.-Y. Chen, and J. Li, "Stochastic modeling of the equilibrium speed-density relationship," *Journal of Advanced Transportation*, vol. 47, no. 1, pp. 126–150, 2013.

[3] A. K. Gupta, S. Sharma, and P. Redhu, "Analyses of lattice traffic flow model on a gradient highway," *Communications in Theoretical Physics*, vol. 62, no. 3, pp. 393–404, 2014.

[4] B. G. Heydecker and J. D. Addison, "Analysis and modelling of traffic flow under variable speed limits," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 2, pp. 206–217, 2011.

[5] X.-L. Ma, D.-F. Ma, D.-H. Wang, and S. Lin, "Modeling of speed-density relationship in traffic flow based on logistic curve," *China Journal of Highway and Transport*, vol. 28, no. 4, pp. 94–100, 2015.

[6] C.-F. Shao, C.-Z. Xiao, B.-B. Wang, and M. Meng, "Speed-density relation model of congested traffic flow under minimum safety distance constraint," *Journal of Traffic and Transportation Engineering*, vol. 15, no. 1, pp. 92–99, 2015.

[7] H. Z. Wang, J. Li, Q.-Y. Chen, and D. Ni, "Logistic modeling of the equilibrium speed-density relationship," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 6, pp. 554–566, 2011.

[8] V. Corcoba Magaña and M. Muñoz-Organero, "WATI: warning of traffic incidents for fuel saving," *Mobile Information Systems*, vol. 2016, Article ID 3091516, 16 pages, 2016.

[9] M.-W. Li, W.-C. Hong, and H.-G. Kang, "Urban traffic flow forecasting using Gauss–SVR with cat mapping, cloud model and PSO hybrid algorithm," *Neurocomputing*, vol. 99, no. 1, pp. 230–240, 2013.

[10] F. Ahmad, I. Khan, S. A. Mahmud et al., "Real time evaluation of shortest remaining processing time based schedulers for traffic congestion control using wireless sensor networks," in *Proceedings of the International Conference on Connected Vehicles and Expo (ICCVE '13)*, pp. 381–391, Las Vegas, Nev, USA, 2013.

[11] H. Y. Shang and Y. Peng, "A new cellular automaton model for traffic flow considering realistic turn signal effect," *Science China Technological Sciences*, vol. 55, no. 6, pp. 1624–1630, 2012.

[12] K. Jung, M. Do, J. Lee, and Y. Lee, "Vehicle running characteristics for interrupted traffic flow by using cellular automata," *The Journal of The Korea Institute of Intelligent Transport Systems*, vol. 11, no. 6, pp. 31–39, 2012.

[13] J. M. del Castillo, "Three new models for the flow-density relationship: derivation and testing for freeway and urban data," *Transportmetrica*, vol. 8, no. 6, pp. 443–465, 2012.

[14] T.-Q. Tang, L. Caccetta, Y.-H. Wu, H.-J. Huang, and X.-B. Yang, "A macro model for traffic flow on road networks with varying road conditions," *Journal of Advanced Transportation*, vol. 48, no. 4, pp. 304–317, 2014.

[15] X. B. Yang, Z. Y. Gao, H. W. Guo, and M. Huan, "Survival analysis of car travel time near a bus stop in developing countries," *Science China Technological Sciences*, vol. 55, no. 8, pp. 2355–2361, 2012.

[16] R. Akçelik, "Relating flow, density, speed and travel time models for uninterrupted and interrupted traffic," *Traffic Engineering and Control*, vol. 37, no. 9, pp. 511–516, 1996.

[17] R. Jiang and Q.-S. Wu, "The traffic flow controlled by the traffic lights in the speed gradient continuum model," *Physica A: Statistical Mechanics and Its Applications*, vol. 355, no. 2–4, pp. 551–564, 2005.

[18] F. J. Wang, W. Wei, H. S. Qi et al., "Research on discontinuous flow speed-density relation model," *Journal of Highway and Transportation Research and Development*, vol. 31, no. 7, pp. 108–114, 2014.

[19] H. Wang, D. Ni, Q.-Y. Chen, and J. Li, "Stochastic modeling of the equilibrium speed-density relationship," *Journal of Advanced Transportation*, vol. 47, no. 1, pp. 126–150, 2013.

[20] F. Soriguera and F. Robusté, "Estimation of traffic stream space mean speed from time aggregations of double loop detector data," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 1, pp. 115–129, 2011.

[21] Y. Lao, G. Zhang, J. Corey, and Y. Wang, "Gaussian mixture model-based speed estimation and vehicle classification using single-loop measurements," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 16, no. 4, pp. 184–196, 2012.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 8586493, 16 pages
http://dx.doi.org/10.1155/2016/8586493

Research Article

A Novel Exercise Thermophysiology Comfort Prediction Model with Fuzzy Logic

Nan Jia,[1] Liang Yu,[1] KaiXing Yang,[1] RuoMei Wang,[1] XiaoNan Luo,[1,2] and QingZhen Xu[3]

[1]*School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China*
[2]*School of Computer and Information Security, Guilin University of Electronic Technology, Guilin, China*
[3]*School of Computer, South China Normal University, Guangzhou, China*

Correspondence should be addressed to XiaoNan Luo; lnslxn@mail.sysu.edu.cn

Received 20 September 2016; Accepted 8 November 2016

Academic Editor: Qingchen Zhang

Participation in a regular exercise program can improve health status and contribute to an increase in life expectancy. However, exercise accidents like dehydration, exertional heatstroke, syncope, and even sudden death exist. If these accidents can be analyzed or predicted before they happen, it will be beneficial to alleviate or avoid uncomfortable or unacceptable human disease. Therefore, an exercise thermophysiology comfort prediction model is needed. In this paper, coupling the thermal interactions among human body, clothing, and environment (HCE) as well as the human body physiological properties, a human thermophysiology regulatory model is designed to enhance the human thermophysiology simulation in the HCE system. Some important thermal and physiological performances can be simulated. According to the simulation results, a human exercise thermophysiology comfort prediction method based on fuzzy inference system is proposed. The experiment results show that there is the same prediction trend between the experiment result and simulation result about thermophysiology comfort. At last, a mobile application platform for human exercise comfort prediction is designed and implemented.

1. Introduction

In the modern society, people are more and more health conscious to improve their life quality. Exercise is often the first step in lifestyle modifications for health maintenance and management [1, 2]. Participation in a regular exercise program can help improve various aspects of cardiovascular function, with reduction in the risk for osteoporosis. Importantly, reductions in risk factors associated with disease states (heart disease, diabetes, etc.) can improve health status and contribute to an increase in life expectancy [3].

During exercise, the human body exchanges energy with the clothing system and environmental conditions in different forms of heat transfer; when the whole human-clothing-environment (HCE) system comes to a thermal steady state, physiological thermal neutrality is reached and the human body will be in a proper thermal and hydration state [4, 5]. A healthy exercise should be thermophysiology comfort for people during exercise. Some parameters can be used to evaluate the thermophysiology comfort, which are temperature, moisture, and physiological properties of heart rate, blood pressure, and so on.

Research on thermophysiology comfort is important for healthy exercise. Based on the thermophysiology comfort model, the human thermal and physiological status can be described and used to predict some accidents like dehydration, exertional heatstroke, syncope, and even sudden death. If these accidents can be analyzed or predicted before they happen during exercise, it will be beneficial to alleviate or avoid disease or mortality. At the Standard Chartered Hong Kong Marathon 2013, 55 runners were reported to have fallen unconscious, to have been rendered comatose, and to have suffered from collapse because of heatstroke [6].

Some landmark research results can be reviewed [7, 8]. Researches about simulating human dynamic thermal comfort in the human-clothing-environment system were presented [9–11]. Wang et al. [9] applied an adaptive neural fuzzy network in the clothing comfort evaluation model. Kingma et al. [10] developed a mathematical model for thermal sensation based on the neurophysiology of thermal

reception. Huizenga et al. [11] reported a model of human physiology and comfort for assessing complex thermal environments. However, limitations have also been identified. There is a lack of enough considerations on the effects of human physiological performances in the thermophysiology comfort prediction. The human body physiological simulation model needs to be enhanced on the dynamic heat and moisture transfer in the human-clothing-environment system.

In this paper, a novel thermophysiology simulation and exercise comfort prediction model is reported. Based on the HCE system, a nonlinear heart rate regulation model and the 25-node thermal regulation model are integrated together to simulate the human physiological performance like temperature, sweat rate, and heart rate. The thermal performance and physiological status of the human body can be simulated in this improved model. Comparisons among different cases show that the improved model can describe the human thermophysiology behavior in the exercise very well. And there is the same prediction trend on the experiment result and simulation result about the thermophysiology comfort.

The main contributions are as follows:

(i) Integrate nonlinear heart rate regulation model into the human thermal physiological simulation model; some important thermophysiology parameters during exercise can be simulated by this integrated model.

(ii) Present a novel exercise thermophysiology comfort prediction model according to the integrated model, which can be used to describe the thermophysiology phenomenon during exercise.

(iii) Implement a mobile application for comfort prediction, in which people get their physiological comfort status according to the exercise information.

The rest of this paper is organized as follows. Related work is introduced in Section 2. An integrated thermal and physiological simulation model in the human-clothing-environment system is reported in Section 3. A novel thermophysiology comfort prediction model with fuzzy logic is presented in Section 4. In Section 5, case studies are designed in different scenes to validate the proposed models. And, in Section 6, a mobile application for comfort prediction is designed and implemented. Finally, conclusion is drawn.

2. Related Work

Research on exercise thermophysiology comfort prediction model involves multidisciplinary knowledge; the human body, clothing, and the environment are a coupled system in the heat and moisture transfer process. The phenomenon of heat and moisture transfer in the HCE system has a significant effect on the human thermophysiology comfort sensation. Figure 1 shows the main components of heat and moisture transfer in HCE system.

From Figure 1, it can be found that the human-clothing-environment system consists of three sets of mathematical models: (1) mathematical description of the thermoregulation of the human body; (2) mathematical description of the heat and moisture transfer processes in clothing; and (3) mathematical description of the coupled heat and moisture transfer processes in the external environment. Based on the simulation results, some thermal and physiological performances can be obtained and used to predict the thermophysiology comfort.

Some research results on heat and moisture transfer in HCE system can be reviewed [12, 13]. Henry proposed mathematical models to simulate the coupled heat and moisture transfer processes in textile fibers [14]. Farnworth developed a numerical solution of the models with the linear assumption [15]. Li et al. further improved the models by incorporating fiber moisture absorption/desorption mechanisms derived from experimental data into the computations [16, 17]. Further development of the mathematical models has taken into account more physical mechanisms such as the liquid water diffusion [18], radiation effects [19], phase change materials [20], pressure gradients [21], and the effect of gravity [22]. The thermal behavior in clothing is simulated.

Mathematical models describing the thermoregulatory system of the human body have been the subject of research for years. Reviewed by Cheng and Fu [23, 24], all the models for the entire body can be characterized in terms of their viewpoint of development. They are (1) one-node models [25], (2) two-node models [26], (3) multinode models [27–29], and (4) multielement models [30, 31]. Though most of them are likely to produce acceptable results under conditions of heat stress when temperature is relatively uniform throughout the body, multinode and multielement models seem to deal better with exposure to cold when large temperature gradients are developed within the body. In the human thermal regulation system, a series of physiological regulatory behaviors (sweating, vasodilatation, shivering, and vasoconstriction) whenever in hot or cold external thermal environments are simulated.

Considering the interactions in the HCE system, the development of a mathematical model of the coupled heat and moisture transfer processes in the external environment is accomplished by the boundary condition equations that refer to the thermal status of the external environment and body [32]. Based on the HCE system, for the given values of humidity, air speed, metabolic rate, and clothing insulation, some simulation results on temperature, moisture, and physiological properties of heart rate and blood pressure can be obtained.

Computer evaluation model is widely used to predict human comfort. Li described the thermophysiological comfort as attainment of a comfortable thermal and wetness state [33]. Wong et al. reported research on neural network predictions of human psychological perceptions of clothing sensory comfort [34] and predicted clothing sensory comfort with artificial intelligence hybrid models [35]. Wang et al. presented the mathematical simulation of the perception of fabric thermal and moisture sensations [36]. Luo et al. presented a fuzzy neural network model for predicting clothing thermal comfort [37]. And Wang et al. designed an adaptive

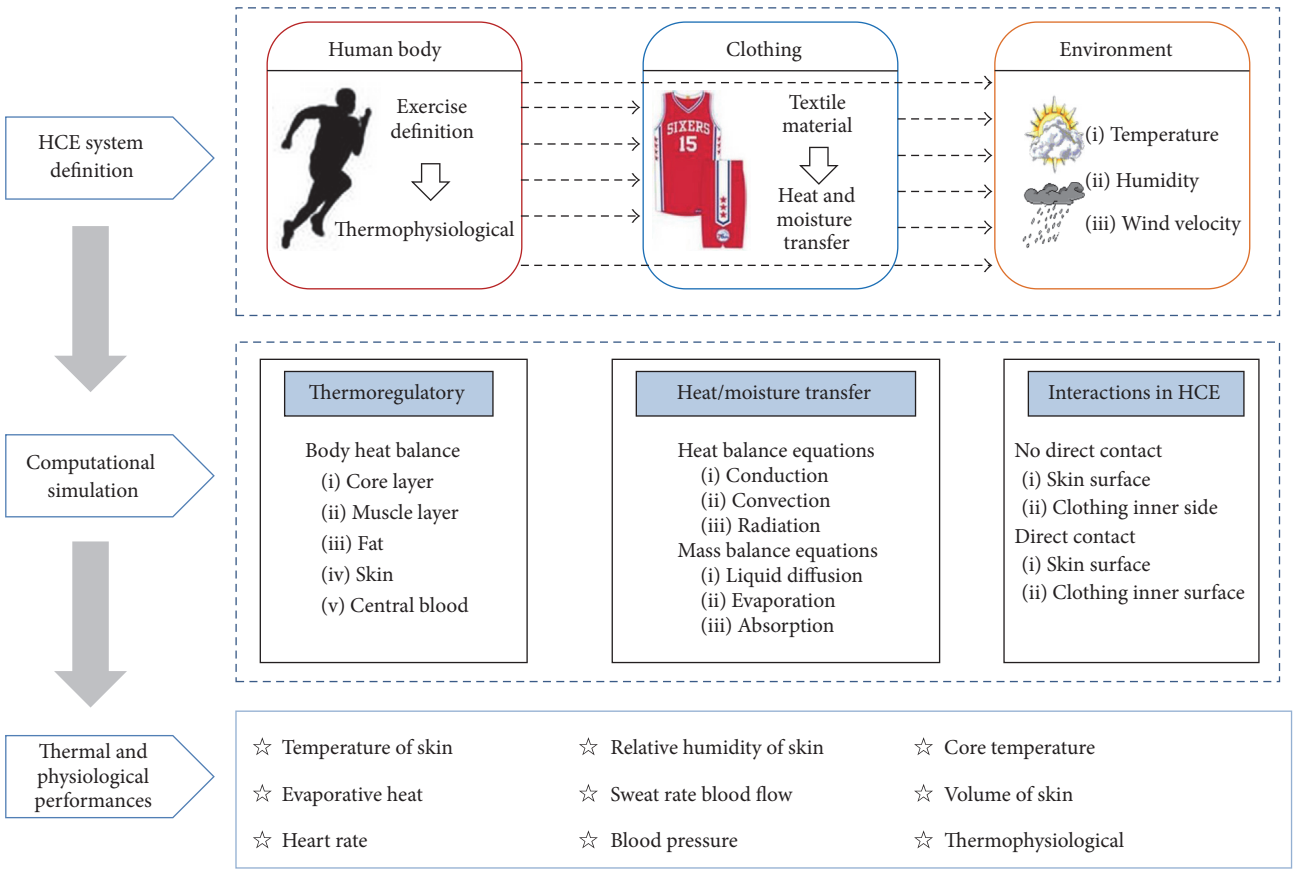


Figure 1: Main components of heat and moisture transfer in HCE system.

neural fuzzy network to build a clothing comfort evaluation model [9].

Although much progress has been made, there are knowledge gaps that need to be filled in individual areas. They are as follows:

(i) There are insufficient advances on the modeling of human body physiological mechanisms during the thermoregulatory processes. Some physiological parameters cannot be simulated in the heat and moisture transfer.

(ii) There is a lack of advances in mathematical modeling of thermophysiology comfort, especially in dynamic heat balance and thermoregulation of a clothed human body.

This paper, therefore, aims to improve the HCE system simulation model and obtain more human body's physiological indicators during exercise, to design a novel exercise thermophysiology comfort prediction model. Figure 2 illustrates the schematic diagram of the thermophysiology comfort prediction model in HCE system reported in this paper.

As shown in Figure 2, the heat and moisture transfer in human-clothing-environment system evokes the effecter mechanisms of the thermoregulatory system to regulate the thermal status of the human body. Based on the 25-node human thermoregulatory model, a nonlinear heart rate regulation model is added to improve human thermal and physiological mode. The improved human thermal and physiological model can be used to describe the human physiological behavior as well as the heart rate regulation behavior during exercise. Many physiological indicators (like core temperature, heart rate, etc.) can be simulated also. According to these simulation results, the fuzzy process of thermophysiology comfort model is used to predict the thermal comfort and health status. The detailed description is shown in the following sections.

3. An Improved Thermal and Physiological Simulation Model in the HCE System

According to the schematic diagram of thermophysiology comfort prediction model shown in Figure 2, it is important to model a reasonable mathematical model to represent the thermal and physiological behaviors in the HCE systems. Considering the feasibility and efficiency of the whole simulation process, we adapt an improved thermal physiological model which comprises a 25-node thermal regulatory model and a nonlinear heart rate regulation model for describing the thermal and physiological regulation system of the human

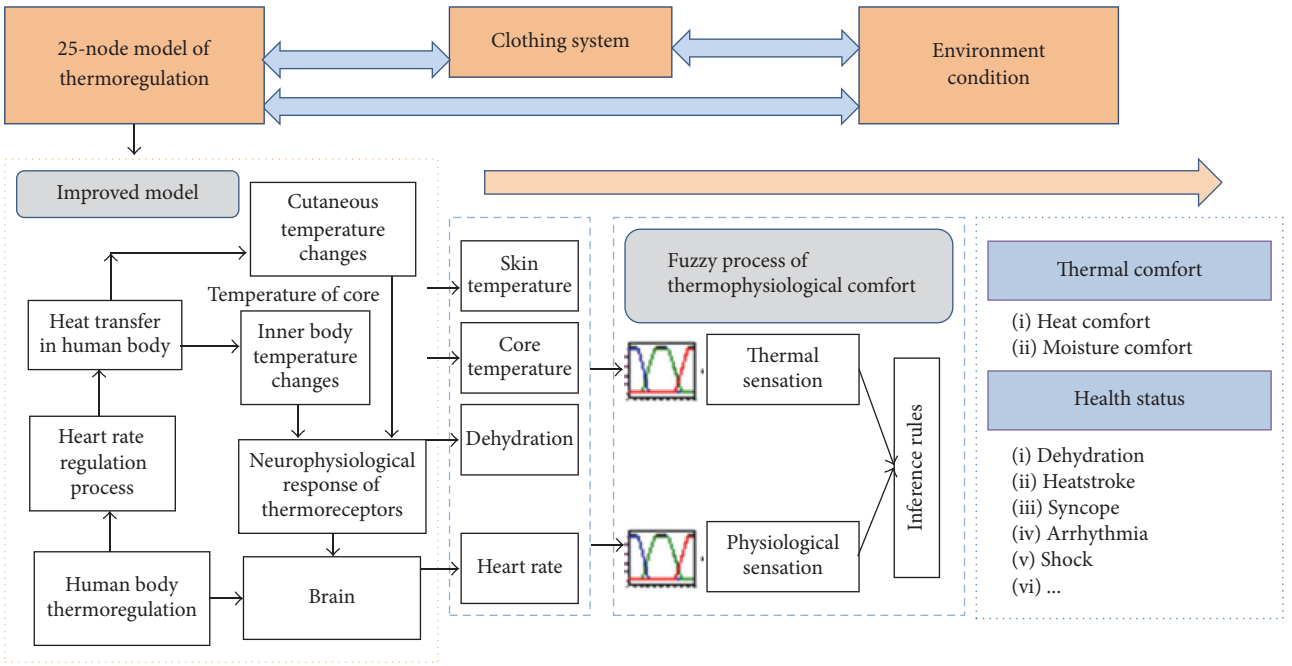


Figure 2: Schematic diagram of thermophysiology comfort prediction model.

body, as well as a coupled heat and moisture transfer model for clothing.

3.1. Improved Thermal Physiological Model of the Human Body. During exercise, the human body activates effective thermoregulatory mechanisms to make the body in a proper thermal status. When the temperature of the human body increases, several physiological reactions are activated automatically to speed up body heat dissipation such as sweating and automatically adjusting the cardiovascular system. During cardiovascular adjustment, the blood is redistributed from the core organs to the skin to facilitate heat dissipation, and the active muscles require blood supply to deliver oxygen for maintenance of activity. Meanwhile, the heart rate (HR) increases to sustain cardiac output and blood supply to the working muscles and the skin [38].

Human thermoregulatory model can be referenced from previous researches [28, 39]. Compared with other models, in this paper, we select 25-node model as the human thermoregulatory model. The more advantages of this model are more accurate physiological simulation performances and efficient numerical computation time cost. In the 25-node model, the human body is divided into six parts: head, trunk, arms, hands, legs, and feet. Each part is expressed by four concentric layers individually representing the core, muscle, fat, and skin layers of the human body, in which all layers are connected by a central blood pool representing large arteries and veins in the body [39]. The thermoregulatory mechanisms of the human body are represented by the mathematical equations

core layer:

$$C(i,1)\frac{dT(i,1)}{dt} = Q(i,1) - B(i,1) - D(i,1) - \mathrm{RES}(i,1), \tag{1}$$

muscle layer:

$$C(i,2)\frac{dT(i,2)}{dt} = Q(i,2) - B(i,2) + D(i,1) - D(i,2), \tag{2}$$

fat layer:

$$C(i,3)\frac{dT(i,3)}{dt} = Q(i,3) - B(i,3) + D(i,2) - D(i,3), \tag{3}$$

skin layer:

$$C(i,4)\frac{dT(i,4)}{dt} = Q(i,4) - B(i,4) + D(i,3) - E(i,4) - Q_t(i,4), \tag{4}$$

central blood:

$$C(25)\frac{dT(25)}{dt} = \sum_{i=1}^{6}\sum_{j=1}^{4} B(i,j), \tag{5}$$

where i is the part number of the human body, j is the layer number in each part, $C(i, j)$ is the thermal capacity of each node (i, j), $C(25)$ is the thermal capacity of central blood, $T(i, j)$ is the temperature of each node, $T(25)$ is the temperature of central blood, $Q(i, j)$ is the metabolic heat generation, $Q_t(i, 4)$ is the dry heat loss on the skin surface, $B(i, j)$ is the thermal exchange between each node and central blood, $D(i, j)$ is the thermal conduction between adjacent layers, $E(i, j)$ is the heat loss by evaporation, and $\mathrm{RES}(i, 1)$ is the heat loss by respiration.

For the control system of model, we have

skin blood:

$$\mathrm{BF}(i,4) = \frac{\mathrm{BFB}(i,4) + \left(SKINV(i) \times D_L\right)}{1 + \left(SKINC(i)\, S_T\right)} \times km(i,4), \quad (6)$$

sweat rate:

$$m_{\mathrm{rsw}}(i) = \frac{\left[C_{\mathrm{sw}}\mathrm{Err}(1,1) + S_{\mathrm{sw}}(\mathrm{Wrms} - \mathrm{Clds}) + P_{\mathrm{sw}}\mathrm{Wrm}(1,1)\,\mathrm{Wrms}\right] \times SKINS(i) \times km(i,4)}{h_{fg}}, \quad (7)$$

where $\mathrm{BF}(i, j)$ is the skin blood flow rate, $\mathrm{BFB}(i, 4)$ is the basal blood flow rate, m_{rsw} is the regulatory sweating rate, C_{sw} and S_{sw} are sweating control coefficients of core and skin, P_{sw} is the overall sweat control coefficient, $SHINC(i)$, $SHINV(i)$, and $SKINS(i)$ are the weighting coefficients of sweating, vasodilatation, and vasoconstriction, D_L and S_T are the vasodilation and vasoconstriction signals, $km(i, 4)$ is the local impact factor, $\mathrm{Err}(i, j)$ is the error signal, $\mathrm{Wrm}(1, 1)$ is the warm signal, Wrms is the integrated warm signal, Clds is the integrated cold signal, and h_{fg} is the evaporation heat of water.

Some important indicators are also presented:

$$\begin{aligned} T_{\mathrm{core}} &= \overline{\sum_{i=1}^{6} T(i,1) + T(25)} \\ T_{\mathrm{skin}} &= \overline{\sum_{i=1}^{6} T(i,4)} \\ M_{\mathrm{s}} &= \int \sum_{i=1}^{6} m_{\mathrm{rsw}}(i)\, dt, \end{aligned} \quad (8)$$

where T_{core} is the mean core temperature of the body, T_{skin} is the mean skin temperature of the body, and M_s is the total sweat accumulation on skin surface.

Just as mentioned above, the cardiovascular system plays a key role in the thermal regulation process. The heart rate is directly affected by the thermoregulatory mechanism. In heart rate regulation, the metabolic rate and the core temperature are two important factors. In this paper, considering the heart rate regulation mechanism and its fluctuating rules, we propose a new heart rate simulation model. This model includes a quadratic function concerning core temperature and a nonlinear term concerning metabolic rate. The nonlinear term is used to simulate the great fluctuation caused by neuroregulation. The equation of the new heart rate regulation model is shown as follows:

$$\mathrm{HR} = N(M) + T\left(T_{\mathrm{core}}\right), \quad (9)$$

where

$$N(M) = \begin{cases} (kM + b)\left(1 - \dfrac{1}{e^{cMt}}\right), & 0 \le t \le t_0 \\ \dfrac{(kM + b)}{\left(e^{d(t - t_0)/M}\right)}, & t > t_0 \end{cases} \quad (10)$$

$$T\left(T_{\mathrm{core}}\right) = p_2 T_{\mathrm{core}}^2 + p_1 T_{\mathrm{core}} + p_0. \quad (11)$$

The functions $N(M)$ and $T(T_{\mathrm{core}})$ account for the effect of the body's nervous regulation system and core temperature regulation on heart rate response, respectively. M denotes the metabolic rate and it is an important indicator to reflect the exercise intensity. T_{core} is the mean core temperature of all core nodes. b, c, d, k, p_2, p_1, and p_0 are function coefficients to be determined, all of which can be estimated in [6, 38].

3.2. Heat and Moisture Transfer Model of Clothing. Clothing plays an important role in providing thermal protection for the human body and creating a portable thermal microclimate between clothing and the human body. The heat and moisture transfer process in clothing is responsible for the temperature and humidity distributions and it directly affects the thermal performance of clothing. Heat conduction, heat convention, heat radiation, moisture absorption/desorption, and so forth are basic heat and moisture transfer ways. In this paper, heat and moisture transfer model of clothing used in the HCE system is referenced by some research reports [32, 33]. The mathematic equations are described in Table 1.

In Table 1, ε_a, ε_f, and ε_l are the volume fractions of water vapor, fibers, and liquid water, respectively. C_a is the water vapor concentration in the air. ρ_l is the density of liquid water in the fibers. C_v is the volumetric heat capacity of fabric. T is the temperature of fabric. K is the effective thermal conductivity of the fabric. D_{fab} is the water vapor diffusion ratio. D_l is the liquid water diffusion ratio. Γ_R is the heat radiation loss. Γ_f is the effective sorption rate of the moisture. Γ_{lg} is the evaporation (condensation) rate of liquid water (vapor). λ_v and λ_l are the heat sorption and desorption of vapor and liquid water. More detailed notations can be found in [6, 39].

Considering the heat and moisture interactions in the HCE system, the human body, clothing, and the environment

Table 1: Heat and moisture transfer equations of clothing.

Vapor moisture	$\frac{\partial (C_a \varepsilon_a)}{\partial t} = \frac{1}{\tau_a} \frac{\partial}{\partial x} \left(D_a \frac{\partial (C_a \varepsilon_a)}{\partial x} \right) - \varpi_1 \Gamma_f + \Gamma_{lg}$
Liquid moisture	$\frac{\partial (\rho_l \varepsilon_l)}{\partial t} = \frac{1}{\tau_l} \frac{\partial}{\partial x} \left(D_l (\varepsilon_l) \frac{\partial (\rho_l \varepsilon_l)}{\partial x} \right) - \varpi_2 \Gamma_f - \Gamma_{lg}$
Heat	$C_v \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(K_{\text{mix}} (x) \frac{\partial T}{\partial x} \right) + \frac{\partial F_R}{\partial x} - \frac{\partial F_L}{\partial x} + \varpi_2 \lambda_v \Gamma_f + \varpi_1 \lambda_v \Gamma_f - \lambda_l \Gamma_{lg}$
Liquid diffusion	$D_l(\varepsilon_l) = \frac{\gamma \cos\theta \sin^2 \alpha d_c {\varepsilon_1}^{1/3}}{20 \eta \varepsilon^{1/3}}$
Condensation	$\Gamma_{lg} = \varepsilon_a h_{lg} S_v (C^* (T) - C_a)$
Moisture sorption	$\Gamma_f = \frac{\partial (C_f \varepsilon_f)}{\partial t} = \frac{1}{r} \frac{\partial}{\partial t} \left(r D_f \frac{(C_f \varepsilon_f)}{\partial r} \right)$
Radiation	$\frac{\partial F_R}{\partial x} = -\beta F_R + \beta \sigma T^4, \ \frac{\partial F_L}{\partial x} = \beta F_L - \beta \sigma T^4$

are a coupled system in heat and moisture transfer. The boundary condition equations of the clothing heat and moisture models are accomplished by reference to the thermal status of the external environment and the body.

In practice, the interactive communications between clothing and the human body and clothing and the environment frequently happen by two boundaries. One is the boundary between the body skin and the inner layer of the clothing close to skin; the other is the boundary between the outer layers of the clothing exposed to the environment [40]. The clothing exchanges energy and moisture with the skin and the external environment, and the thermal status and physiological status are automatically updated.

For the inner side of the clothing close to the skin,

$$\begin{aligned} -\frac{D_a}{\tau_a} \frac{\partial (C_a \varepsilon_a)}{\partial x}\bigg|_{i,x=0} &= \frac{p_m E(i,4)}{\left(\lambda_{lg} S(i)\right)} \\ -\frac{D_l}{\tau_l} \frac{\partial (\rho_l \varepsilon_l)}{\partial x}\bigg|_{i,x=0} &= \kappa_2 \lambda_{lg} h_{lg} \left({C_a}^* \left(T_{cl,0} \right) - C_{ask} (i) \right) \\ -K_{\text{mix}} \frac{dT}{dx}\bigg|_{i,x=0} &= p_m \left(H_{t1} \left(T_{cl,0} - T(i,4) \right) \right) + \frac{p_h E(i,4)}{S(i)} \\ &\quad + \kappa_2 \lambda_{lg} h_{lg} \left({C_a}^* \left(T_{cl,0} \right) - C_{ask} (i) \right). \end{aligned} \tag{12}$$

For the outer side of the clothing exposed to the environment,

$$\begin{aligned} -\frac{D_a}{\tau_a} \frac{\partial (C_a \varepsilon_a)}{\partial x}\bigg|_{i,x=L} &= \kappa_1 H_{m2} \left(C_{acl,L} - C_{\text{env}} \right) \\ -\frac{D_l}{\tau_l} \frac{\partial (\rho_l \varepsilon_l)}{\partial x}\bigg|_{i,x=L} &= \kappa_2 h_{lg} \left({C_a}^* \left(T_{cl,L} \right) - C_{\text{env}} \right) \\ -K_{\text{mix}} \frac{dT}{dx}\bigg|_{i,x=L} &= \kappa_2 \lambda_{lg} h_{lg} \left({C_a}^* \left(T_{cl,L} \right) - C_{a\text{env}} (i) \right) \\ &\quad + H_{t2} \left(T_{cl,L} - T_{\text{env}} \right), \end{aligned} \tag{13}$$

where p_h and p_m are the proportions of moisture vapor and dry heat loss from skin at the clothing-covered area; κ_1 and κ_2 are the transfer proportions of water vapor and liquid water; H_t is the heat conduction coefficient of air; H_m is the mass transfer coefficient.

4. An Exercise Thermophysiology Comfort Prediction Model

Human comfort can be used to describe the overall state of the body physiologically, which is an important index of body wellbeing. Current researches on human comfort mainly focus on the unilateral prediction of thermal comfort. But, in reality, human thermal senses directly affect the physiological changes. For example, as the temperature of the human body rises, the heart rate, blood pressure, and other physiological signs will change as well. Therefore, the thermal comfort and physiological comfort should be integrated and taken into account. In this paper, an exercise thermophysiology comfort prediction model is designed. The fuzzy inference system [9] is used in the comfort prediction model. Some thermal and physiological indicators during exercise can be used as the input to predict the thermal comfort and health status.

For the various simulated indicators in our thermal physiological model, we select mean skin temperature, mean core temperature, and change rate of mean skin temperature as input variables, and the prediction results of thermal comfort will be got after the reasoning process. Correspondingly, we select the mean core temperature, sweat accumulation (it is approximately equal to the amount of dehydration), and heart rate as input variables and evaluate the physiological comfort. The comfort variables and the related fuzzy sets are listed in Table 2.

Equation (14) is the trapezoidal function, which is used to define the membership function of every input. The feeling interval for different thermal and physiological sensation can be obtained based on the trapezoidal function:

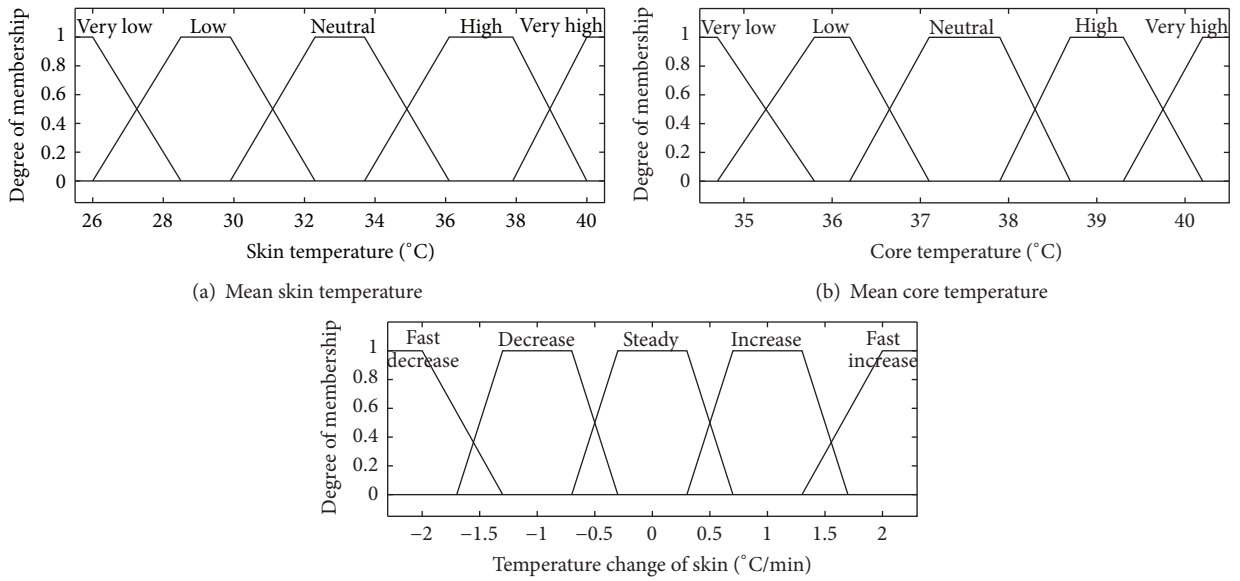
$$\mu(x) = \begin{cases} \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c \le x \le d, \end{cases} \tag{14}$$

where a, b, c, and d are threshold parameters used to determine the shape of the membership function. By changing these parameters, the feeling interval of thermal and physiological status can be well defined and divided.

Figures 3 and 4 show the membership functions of thermal and physiological comfort inputs, respectively. All the

Table 2: The list of comfort variables and the related fuzzy sets.

Name	Variables	Fuzzy sets
Thermal comfort S_l	T_{skin}	Very low, low, neutral, high, very high
	T_{core}	Very low, low, neutral, high, very high
	dT_{skin}/dt	Fast decrease, decrease, neutral, increase, fast increase
Physiological comfort S_p	T_{core}	Very low, low, neutral, high, very high
	M_s	Severe dehydration, moderate dehydration, slight dehydration, normal
	HR	Low, normal, high
Overall comfort S	S_l	Cold, cool, neutral, warm, hot
	S_p	High risk, low risk, normal
	S	Uncomfortable, acceptable, comfortable



(a) Mean skin temperature

(b) Mean core temperature

(c) Change rate of mean skin temperature

Figure 3: The membership function of thermal comfort indicators.

threshold parameters are given by experts on sports medicine [41].

According to the human mean skin temperature, mean core temperature, and change rate of mean skin temperature, we define a thermal comfort function to predict the human thermal status during exercise. In (15), S_l is the thermal comfort sensation and f is the thermal comfort inference function, in which an adaptive neurofuzzy inference system (ANFIS) is introduced to evaluate the thermal comfort. The ANFIS achieves fuzzification, fuzzy reasoning, and defuzzification process using a neutral work, while taking advantage of the information storage capacity and learning ability of artificial neural network [9]. Hence,

$$S_l = f\left(T_{\text{skin}}, T_{\text{core}}, \frac{dT_{\text{skin}}}{dt}\right). \tag{15}$$

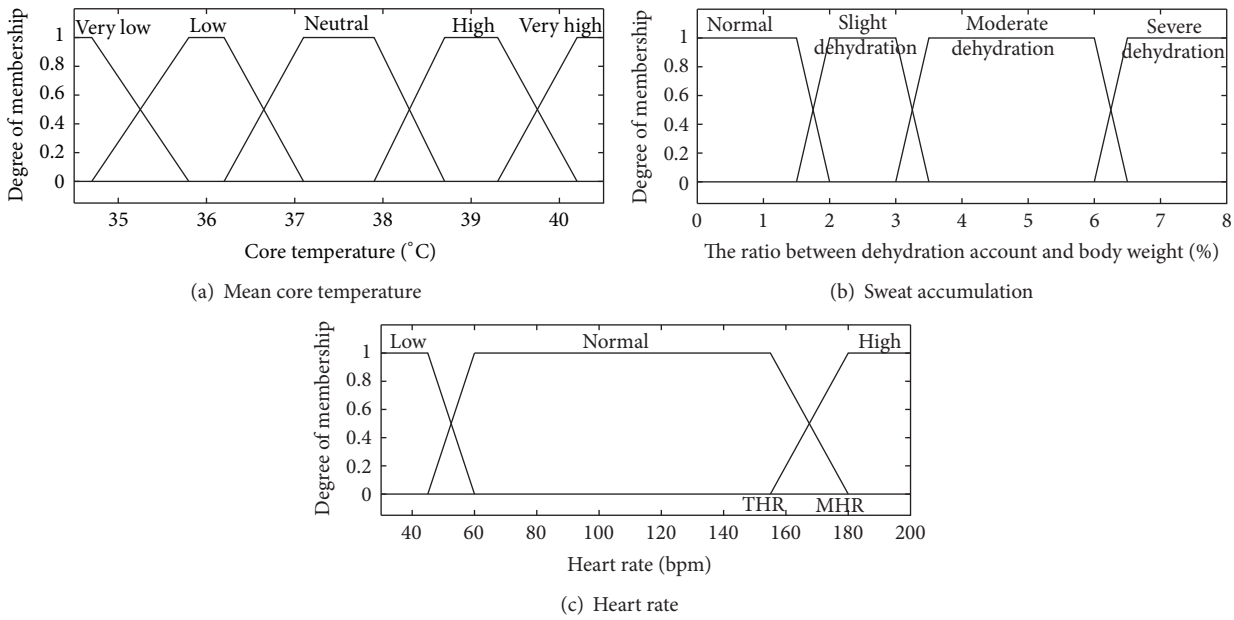
At the same time, physiological comfort of the human body should be considered. In accordance with the human mean core temperature, sweat accumulation, and heart rate, we define a physiological sensation function to predict the human health status during exercise. Equation (16) is the definition of physiological comfort sensation. In the physiological comfort inference process, we also construct the ANFIS to obtain the physiological comfort sensation. Hence,

$$S_p = f\left(T_{\text{core}}, M_s, \text{HR}\right). \tag{16}$$

Under a series of comfort inference rules, the overall comfort in (17) can be calculated from the thermal comfort sensation and physiological comfort sensation. All the rules defined here are based on a large number of statistical analyses and medicine knowledge [6, 33, 41]. Some representative inference rules defined in this paper are presented as follows:

$$S = R\left(S_l, S_p\right). \tag{17}$$

Rule 1. If thermal sensation is neutral and physiological sensation is normal, then overall comfort is comfortable.



(a) Mean core temperature

(b) Sweat accumulation

(c) Heart rate

Figure 4: The membership function of physiological comfort indicators.

Table 3: Cases definition.

Case	Settings									
	Subject			Clothing		Environment		Exercise		
	Gender	Age (years)	Weight (kg)	Body area (m^2)	Composition	Coverage rate	Temperature (°C)	Relative humidity	Exercise type	Duration (min)
Case 1	Male	25	70	1.8	98% cotton, 2% lycra	70%	32	50%	Running	30
Case 2									Jogging	30
Case 3									Walking	30

Rule 2. If thermal sensation is cool and physiological sensation is normal, then overall comfort is acceptable.

Rule 3. If thermal sensation is hot and physiological sensation is high risk, then overall comfort is uncomfortable.

5. Case Study

Three exercise cases are designed to evaluate the exercise thermophysiology comfort prediction model.

5.1. Scenes Setting. Different types of exercises are used for thermal physiological simulation and comfort prediction. Table 3 shows the cases definition. In the three cases, the subject is the same person (25 years old, 70 kg, about 1.8 m^2 body surface area). Also, the subject wears the same cotton suit with 70% coverage rate. The environment temperature and relative humidity are set to 32°C and 50%. To set up several different scenes, three types of exercises for 30 minutes are introduced, that is, running, jogging, and walking.

5.2. Results Discussion

5.2.1. Simulation Results of Thermal Physiological Model. During exercise, the human body's thermal status and physiological status dynamically change, mainly reflected in the following phenomenon: temperature rising, heart rate accelerating, sweating increasing, and so forth. Figure 5 shows the simulated tendency curves of four important physiological indicators. It illustrates that the values of human physiological indicators are changed to varying degrees in different exercises. The higher the exercise intensity we choose, the more significant the changes in simulation.

Figure 5(a) illustrates the changes of mean core temperature in three scenes. Because of the thermoregulatory behaviors such as metabolic heat production, exercise heat production, and sweating, the core temperature increases gradually and it is maintained in balance within a certain temperature. Figure 5(b) illustrates the changes of mean skin temperature and Figure 5(c) illustrates the sweat accumulation of the human body. As shown in Figure 5(b), the values of mean skin temperature increase rapidly first and

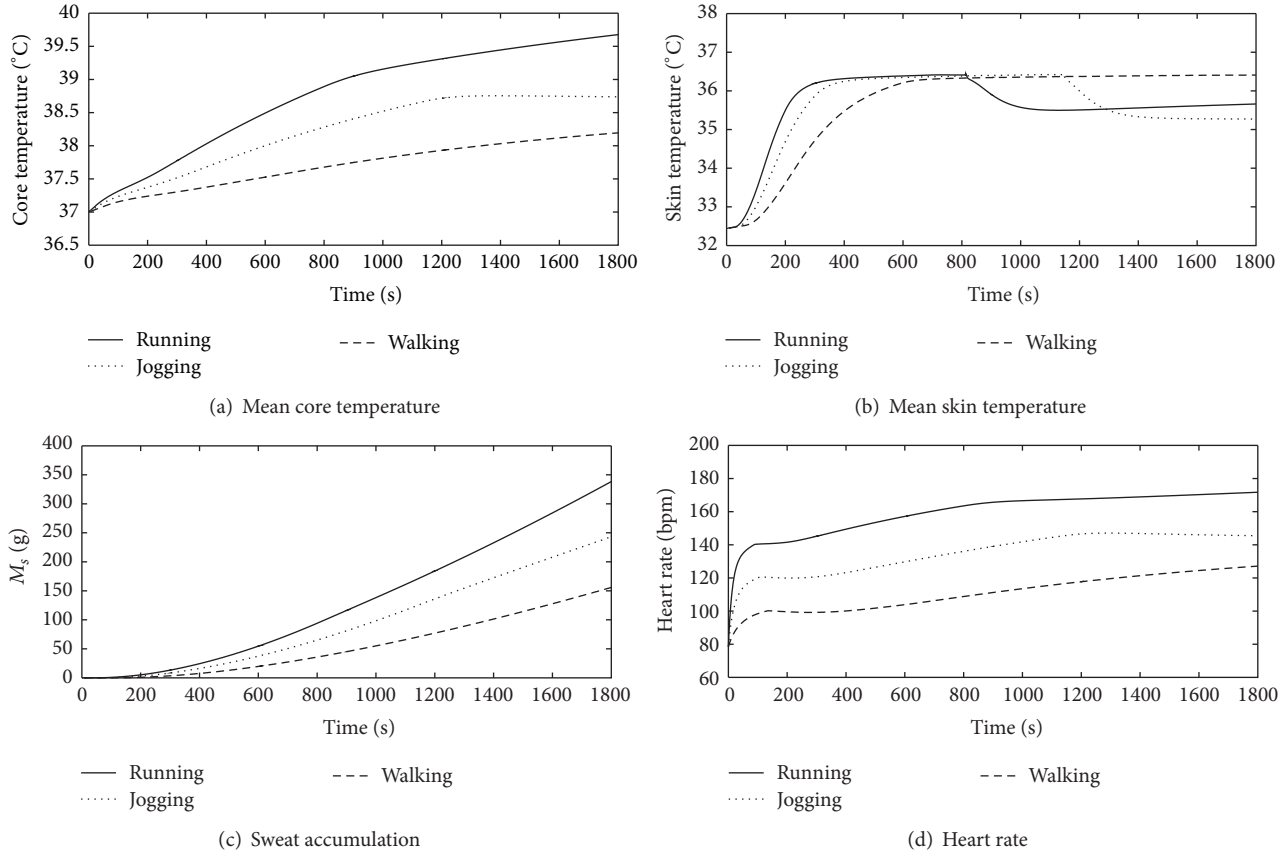


FIGURE 5: The simulation results of our improved thermal physiological model in different scenes; they should be listed as (a) mean core temperature tendency, (b) mean skin temperature tendency, (c) sweat accumulation tendency, and (d) heart rate tendency.

then keep balance for a short time (especially in jogging and running) and at last decline to secondary balance state, while a relatively high skin temperature in walking continues for a long time. The variation of mean skin temperature is mainly due to the sweat evaporation of the human body, which can bring out heat from skin and decrease the skin temperature. Since the sweat in running is greater than that in jogging (reflected in Figure 5(c)), the mean skin temperature curve in running fell earlier than that in jogging in Figure 5(b). Meanwhile, the accumulation of sweat in walking affects the skin temperature slightly; thus, the tendency curve in walking increases first and then keeps balance. Figure 5(d) presents the changes of heart rate. As the heart rate simulation is directly associated with the core temperature and metabolic rate, its distribution is in agreement with the core temperature curves in Figure 5(a); running achieves the highest heart rate values, jogging is the second, and walking is the lowest.

5.2.2. Thermophysiology Comfort Prediction. Table 4 shows the simulated values and overall comfort in three different scenes. In general, walking makes people feel comfortable and acceptable in the whole process, while jogging makes people feel comfortable and acceptable in a relatively long period of time, and running makes people feel uncomfortable in a relatively long period of time.

In the first 166 s of case 1, people feel comfortable, since all thermal sensation and physiological sensation keep normal. From 167 s to 513 s, people feel acceptable since thermal sensation changes to warm, but physiological sensation still keeps normal. After that, from 514 s, both the thermal values and the physiological values are changed; people feel uncomfortable with the thermal sensation getting into hot and physiological sensation getting into low risk and even high risk. In case 2, from 0 s to 213 s, people feel comfortable since both thermal sensation and physiological sensation are normal. From 214 s to 815 s, the human comfort is acceptable. From 816 s, people feel uncomfortable with the thermal sensation getting into hot and physiological sensation getting into low risk. In case 3, from 0 s to 320 s, people feel comfortable. After that, people feel acceptable until the end of the exercise.

The experiment results show some important and valuable suggestions: for example, walking is a comfortable and acceptable exercise in daily life; our simulated results also tell us that walking within 30 minutes is acceptable and cannot cause any discomfort. Jogging for a relatively long period of time also makes people feel comfortable, while it will make people feel uncomfortable when the exercise time exceeds 13.5 minutes. Therefore, we should pay attention to drinking water and cooling while jogging for a long time. The results also show that running will easily cause body discomfort. When people run at 32°C in 8 minutes, it is easy

Table 4: Results of thermophysiology comfort prediction for three cases.

Cases	Time (seconds)	Thermal values T_{skin} /T_{core}/$dT_{skin}/dt/S_l$ (°C/°C/°C/—)	Physiological values T_{core}/M_s/HR/S_p (°C/g/—/—)	Overall comfort
Case 1	$t = 0$	32.44/36.99/0.44/neutral	36.99/0.00/78.49/normal	Comfortable
	$t = 166$	34.89/37.44/2.89/neutral	37.44/3.22/140.95/normal	Comfortable
	$t = 167$	34.91/37.45/2.91/warm	37.45/3.27/140.97/normal	Acceptable
	$t = 513$	36.36/38.29/4.36/warm	38.29/40.22/153.93/normal	Acceptable
	$t = 514$	36.36/38.30/4.36/hot	38.30/40.37/153.97/low risk	Uncomfortable
	$t = 1800$	35.66/39.67/3.66/hot	39.67/338.35/171.69/high risk	Uncomfortable
Case 2	$t = 0$	32.44/36.99/0.44/neutral	36.99/0.00/78.49/normal	Comfortable
	$t = 213$	34.88/37.38/2.88/neutral	37.38/3.52/119.90/normal	Comfortable
	$t = 214$	34.90/37.39/2.90/warm	37.39/3.57/119.90/normal	Acceptable
	$t = 815$	36.38/38.29/4.38/warm	38.29/67.36/136.49/normal	Acceptable
	$t = 816$	36.38/38.30/4.38/hot	38.30/67.51/136.52/low risk	Uncomfortable
	$t = 1800$	35.27/38.73/3.27/hot	38.73/243.23/145.43/low risk	Uncomfortable
Case 3	$t = 0$	32.44/36.99/0.44/neutral	36.99/0.00/78.49/normal	Comfortable
	$t = 320$	34.89/37.31/2.89/neutral	37.31/4.16/99.28/normal	Comfortable
	$t = 321$	34.90/37.31/2.90/warm	37.31/4.20/99.29/normal	Acceptable
	$t = 1800$	36.41/38.19/4.41/warm	38.19/155.83/127.12/normal	Acceptable

for them to feel uncomfortable. Running makes people feel uncomfortable by changing body temperature into a hot state and putting the human body's physiological state into a high risk state. So, we recommend not to run for a long time in a hot environment.

5.3. More Discussion. The aim of the case study is to evaluate human comfort under different exercise intensities. Therefore, we take exercise intensity as variable, and other factors (subject, clothing, environment, etc.) as invariants in the setting of the case study. It is worth noting that this does not mean that our model cannot simulate the thermal physiological changes and predict human comfort caused by other factors. To support this conclusion, some extra cases are discussed as follows.

(i) Subject. To validate that our model is subject-sensitive, subjects of different gender, age, and body type have been chosen and series cases are designed and simulated. From the simulation results, it can be concluded that different personal parameter settings can affect the human thermoregulatory mechanism and cause different degrees of thermal physiological and human comfort change. For example, core temperature, sweat accumulation, and heart rate in old people are lower than those in young people. Besides, old people are more likely to feel uncomfortable. Parts of the simulated tendency curves such as mean skin temperature and sweat accumulation are shown in Figure 6. Therefore, our simulation is subject-sensitive.

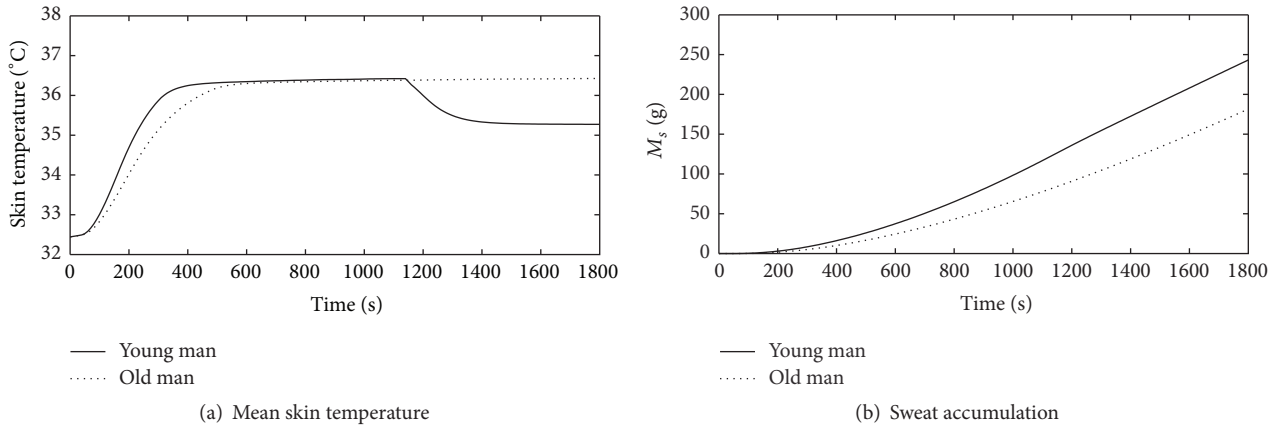
(ii) Clothing. To validate that our model is clothing-sensitive, a series of contrast experiments with different clothing settings are conducted, and parts of the simulated tendency curves such as mean core temperature and sweat accumulation are shown in Figure 7.

(iii) Environment. To validate that our model is environment-sensitive, we also conduct a series of contrast experiments with different environment settings. The result shows that a man running in an extreme environment like high temperature and high humidity can easily experience discomfort both in thermal sensation and in physiological sensation. Figure 8 shows the tendency curves and Table 5 lists the predicted comfort.

Although we have not elaborated the effects of subject, clothing, and environment on human thermal physiological simulation as well as comfort prediction, our thermal physiological model and comfort prediction model are capable of simulating and analyzing the effects caused by these factors in HCE systems.

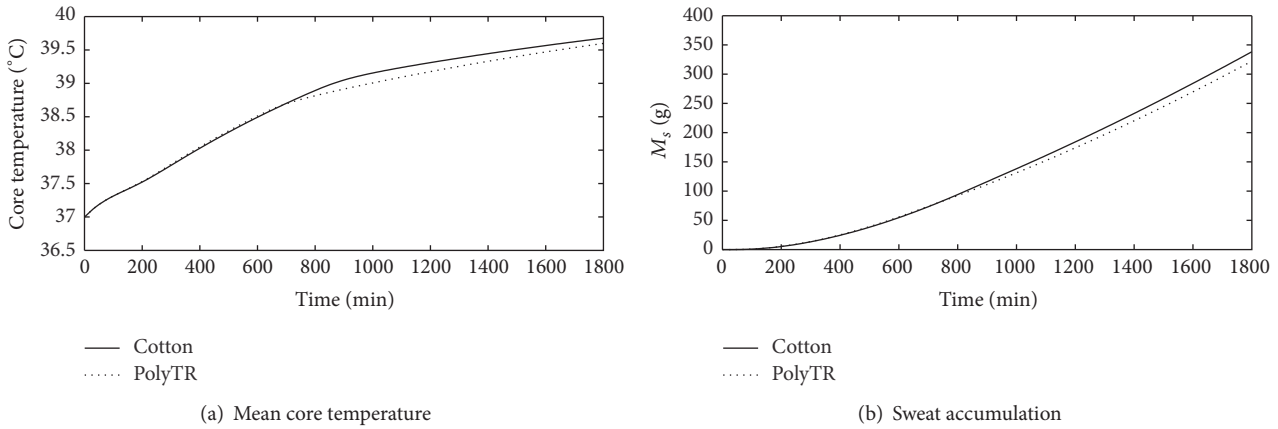
6. A Mobile Application for Human Comfort Prediction

With the development of mobile communication technology and increasing popularization of the Internet, mobile multimedia services are more and more favored by users. Various mobile devices and applications are designed to aid people to improve their life quality. Exercise thermophysiology comfort is regarded as one of the most important and significative research areas, which has been focused upon in recent years. According to the previous description and the mobile application requirements of human comfort, a user-friendly smart application with low computational requirements has been developed to evaluate human exercise comfort in daily



(a) Mean skin temperature (b) Sweat accumulation

Figure 6: The tendency curves of mean skin temperature and sweat accumulation with different subjects.

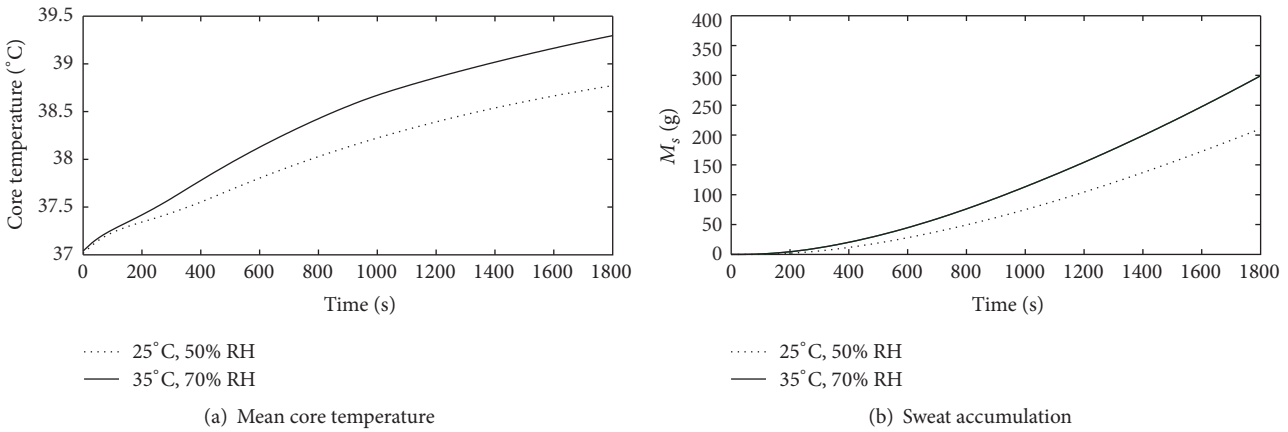


(a) Mean core temperature (b) Sweat accumulation

Figure 7: The tendency curves of mean core temperature and sweat accumulation with different clothing.

Table 5: Results of thermophysiology comfort prediction in different environments.

Cases	Time (seconds)	Thermal values $T_{skin}/T_{core}/dT_{skin}/dt/S_l$ (°C/°C/°C/—)	Physiological values $T_{core}/M_s/\text{HR}/S_p$ (°C/g/—/—)	Overall comfort
25°C, 50% RH	$t = 0$	31.67/37.01/−0.32/neural	37.01/0.00/79.19/normal	Comfortable
	$t = 302$	34.90/37.44/2.90/neural	37.44/5.88/117.47/normal	Comfortable
	$t = 303$	34.90/37.44/2.91/warm	37.44/5.93/117.48/normal	Acceptable
	$t = 1087$	35.92/38.30/3.92/warm	38.30/87.40/133.78/normal	Acceptable
	$t = 1088$	35.92/38.30/3.92/hot	38.30/87.55/133.80/low risk	Uncomfortable
	$t = 1800$	35.96/38.77/3.96/hot	38.77/210.52/146.46/low risk	Uncomfortable
35°C, 70% RH	$t = 0$	32.88/37.04/0.88/neutral	37.04/0.00/80.31/normal	Comfortable
	$t = 162$	34.89/37.36/2.89/neutral	37.36/2.66/121.68/normal	Comfortable
	$t = 163$	34.91/37.36/2.91/warm	37.36/2.70/121.68/normal	Acceptable
	$t = 713$	36.60/38.30/4.60/warm	38.30/61.56/138.11/normal	Acceptable
	$t = 714$	36.60/38.30/4.60/hot	38.30/61.72/138.14/low risk	Uncomfortable
	$t = 1800$	36.60/39.30/4.60/hot	39.30/299.33/160.91/low risk	Uncomfortable



(a) Mean core temperature

(b) Sweat accumulation

Figure 8: The tendency curves of mean core temperature and sweat accumulation with different environments.

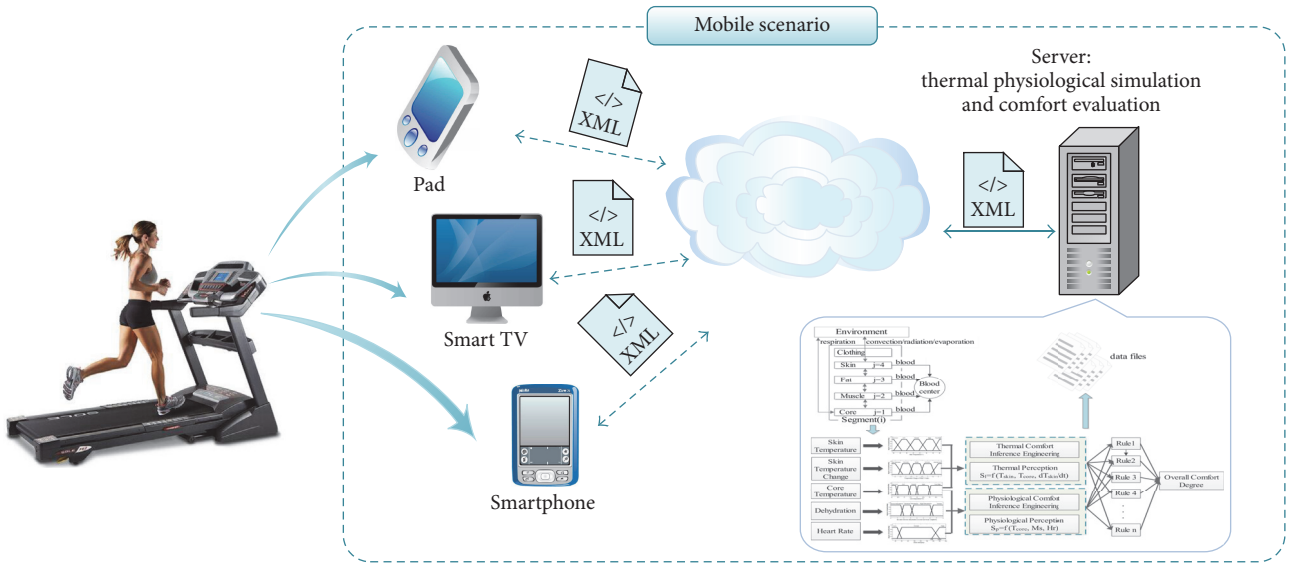
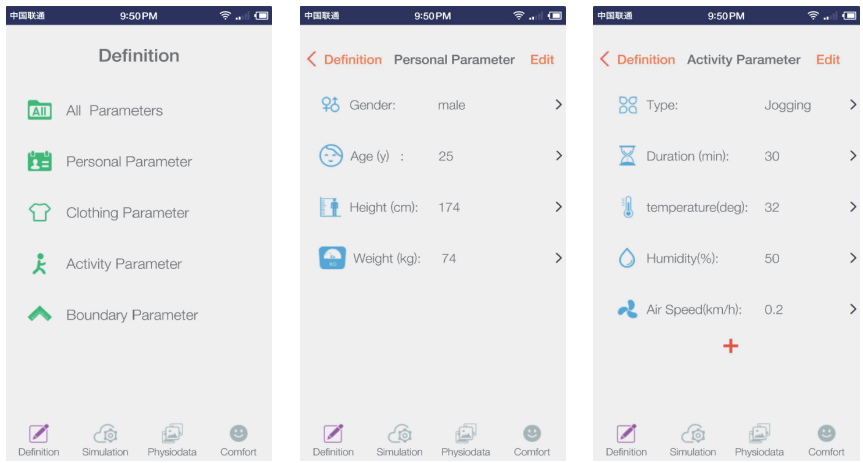


Figure 9: The basic architecture for the comfort prediction prototype in mobile scenario.

life, which allows easily changing the simulation scenes and simulating the human physiological status as well as carrying out comfort evaluation and prediction. The basic architecture for the prototype is shown in Figure 9. Through a variety of mobile devices, the scene parameters are set and transmitted to the server side to compute the comfort sensation of the human body.

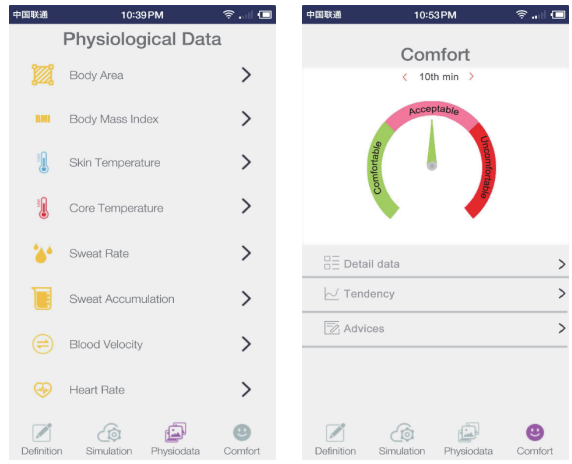
6.1. App Input: Scene Parameters Definition. Various parameters in the scene of case study directly affect simulation results, and the different combination of scene parameters will produce different physiological state and comfort sensation. Four main types of scene parameters are defined, which are personal parameter, clothing parameter, activity parameter, and boundary parameter. Figure 10(a) shows the list view of scene parameters, and Figures 10(b) and 10(c) are the detailed views. The personal parameter includes the gender, age, height, weight, and some specific human physiological parameters like the density of blood and specific heat of the body. Because these physiological parameters have small variations among individuals, they are preset with default values. If necessary, these specific physiological parameters can be input by customs themselves. The clothing parameter includes the clothing style, composition, and coverage rate. Besides, the fabric parameters like porosity, capillary angle, and heat transfer coefficients are also preset. The activity parameter contains environment and exercise settings. Temperature, humidity, wind velocity, and exercise type and duration are all considered. The boundary parameter defines some interactive information of HCE systems, such as skin temperature and inner garment temperature.

6.2. Server: Physiological Simulation and Comfort Prediction. Server side is used to handle the time-consuming and computing resource-intensive simulation task in HCE system. It takes scene parameters as input and outputs the body comfort sensation.



(a) Scene definition (b) Personal parameter (c) Activity parameter

Figure 10: The scene definition views of the app.



(a) Physiological data list (b) Comfort result

Figure 11: The views of simulation results.

The server receives input parameters from clients, and then numerical solutions are taken to solve the human physiological model, heat and moisture transfer model, and the interactive equations between body and clothing. After that, the server uses the simplified neurofuzzy inference system to carry out comfort evaluation and prediction. The simulated results such as human skin temperature, heart rate, sweat rate, and comfort sensation are generated. At last, all these results are transferred back to the smart devices. The data transmission between the client and the server is encapsulated as a customized file in XML format.

6.3. App Output: Results Visualization. A variety of graphical representations are used in our application to help users visualize the change of physiological state and comfort sensation. Figure 11 shows the visualization screenshots of the app (corresponding to "Physiodata" in Figure 11(a) and "Comfort" in Figure 11(b)). Figure 11(a) shows the calculated physiological data, and Figure 11(b) displays the overall comfort sensation. The detailed information of comfort sensation can be obtained, and also some pieces of advice related to exercise comfort sensation are given.

7. Conclusion

During exercise, the heat balance of the human body is maintained by the processes of heat production and heat loss via radiation, conduction, convection, and evaporation. These processes would have effects on the physiological responses and influence the thermal status and comfort perception. In order to predict human thermophysiology comfort, in this paper, a heart rate regulation model is added to HCE system to simulate the human body thermal physiological behavior; according to this improved model, some important physiological parameters can be obtained. Further, in this paper, a novel thermophysiology comfort prediction model and a user-friendly mobile application for human comfort prediction are designed. The experiment results show that

there is the same prediction trend on the experiment result and simulation result about thermophysiology comfort. The proposed exercise thermophysiology comfort prediction model still has some limitation. The thermal physiological mechanism needs to be researched to simulate the human physiological sensation further. We need to achieve and analyze more exercises and even investigate how to apply this simulation model and comfort model in the health services.

Nomenclature

β: Radiation absorption constant of the fiber (m^{-1})
ε: Porosity of the fabric
ε_a: Volume fraction of water vapor
ε_f: Volume fraction of fibers
ε_l: Volume fraction of liquid phase
η: Dynamic viscosity of liquid (kg/ms)
γ: Surface tension of fiber (J/m)
Γ_f: Effective sorption rate of the moisture
Γ_{lg}: Evaporation/condensation rate of the liquid/vapor
λ_l: Heat of sorption or desorption of liquid by fibers (kJ/kg)
λ_v: Heat of sorption or desorption of vapor by fibers (kJ/kg)
ρ_l: Density of the liquid water (kg/m^3)
σ: Stefan-Boltzmann constant (W/m^2 K)
τ_a: Effective tortuosity of the fabric for water vapor diffusion
θ: Contact angle of the liquid water on the fiber surface
$B(i, j)$: Heat transfer by blood flow in node (i, j) (W)
$\mathrm{BF}(i, j)$: Blood flow rate of node (i, j) (l/h)
$\mathrm{BFB}(i, j)$: Basal blood flow rate of node (i, j) (l/h)
$C(25)$: Thermal capacity of the blood (Wh/°C)
$C(i, j)$: Thermal capacity in node (i, j) (Wh/°C)
$C^*(T)$: Saturated water vapor concentration at T (kg/m^3)
C_a: Water vapor concentration in the air filling the interfiber void space (kg/m^3)
$C_h(i, j)$: Shivering metabolic heat generation in node (i, j) (W)
C_v: Volumetric heat capacity of the fabric (kJ/m^3 K)
$Chilf(i)$: Weighting and distribution coefficient of shivering muscles
$\mathrm{Cld}(i, j)$: Cold signal of node (i, j)
Clds: Integrated cold signal of the whole skin surface
$D(i, j)$: Heat transfer by thermal conduction in node (i, j) (W)
D_a: Diffusion coefficient of water vapor in the air of the fabric (m^2/s)
D_f: Diffusion coefficient of water vapor in the fibers of the fabric (m^2/s)
D_L: Control signal of vasodilation
D_l: Diffusion coefficient of liquid water in the fibers of the fabric (m^2/s)
$E(i, j)$: Heat loss by evaporation through the skin surface in node (i, j) (W)
$\mathrm{Err}(i, j)$: Error signal of node (i, j)
$F_{L/R}$: Elementary total thermal radiation incident inside the clothing travelling to the left/right (W/m^2)
h_c: Convection heat transfer coefficient (W/m^2 K)
h_r: Radiation heat transfer coefficient ($W/m^2 \cdot °C$)
$h_t(i)$: Integrated heat transfer coefficient ($W/m^2 \cdot °C$)
h_{fg}: Evaporation heat of water (J/kg)
$h_{l\leftrightarrow g}$: Mass transfer coefficient for evaporation and condensation (m/s)
k_a: Thermal conductivity of the air ($mmW/m^2 \cdot °C$)
K_{mix}: Effective thermal conductivity of the fabric (W/m/K)
$km(i, 4)$: Regional influence factor
$m_s(i)$: Sweating accumulation on the skin surface in the ith part (g/m^2)
$m_{\mathrm{rsw}}(i)$: Regulatory sweating in the ith part ($g/s/m^2$)
$P_{\mathrm{ea}}(i)$: Water vapor pressure of ambient temperature in the ith part (Pa)
$P_{\mathrm{sat}}(i)$: Saturation water vapor pressure on the skin temperature in the ith part (Pa)
$P_{\mathrm{sk}}(i)$: Water vapor pressure on the skin surface in the ith part (Pa)
$Q(i, j)$: Metabolic heat generation in node (i, j) (W)
$Q_b(i, j)$: Basal metabolic heat generation in node (i, j) (W)
$Q_t(i, j)$: Heat loss by convection and thermal radiation in node (i, j) (W)
r: Radius (mm)
$R_{\mathrm{ea}}(i)$: Evaporation heat resistance on the skin surface in the ith part (m^2 Pa/W)
$R_{\mathrm{esk}}(i)$: Evaporation resistance of the skin in the ith part (m^2 Pa/W)
$\mathrm{RES}(i, 1)$: Latent respiration heat loss in node $(i, 1)$ (W)
$RT(i, 4)$: Width of temperature
S_T: Control signal of vasoconstriction
S_v: Surface-to-volume ratio of the fiber (m^{-1})
$SKINC(i)$: Weighting and distribution coefficient of vasoconstriction in the ith part
$SKINR(i)$: Integrated weight coefficient
$SKINS(i)$: Weighting and distribution coefficient of sweating in the ith part
$SKINV(i)$: Weighting and distribution coefficient of vasodilation in the ith part
T: Temperature of the fabric (K)
$T(25)$: Temperature of the blood (°C)
$T(i, j)$: Temperature of node (i, j) (K)

t_{aI}: Thickness of the air layer (mm)
$T_{set}(i, j)$: The set-point temperature of node (i, j) (°C)
$W(i, j)$: Work accomplished in node (i, j) (W)
Wrm(i, j): Warm signal of node (i, j)
Wrms: Integrated warm signal of the whole skin surface.

Competing Interests

The authors declare no competing interests.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (NSFC) (nos. 61320106008, 61402185, and 61672547).

References

[1] K. R. Miller, S. A. McClave, M. B. Jampolis et al., "The health benefits of exercise and physical activity," *Current Nutrition Reports*, vol. 5, no. 3, pp. 204–212, 2016.

[2] E. Anderson and G. Shivakumar, Effects of exercise and physical activity on anxiety. Progress in Physical activity and Exercise and Affective and Anxiety Disorders: Translational Studies, Perspectives and Future Directions, 2015.

[3] S. M. Phillips, T. R. Wójcicki, and E. McAuley, "Physical activity and quality of life in older adults: an 18-month panel analysis," *Quality of Life Research*, vol. 22, no. 7, pp. 1647–1654, 2013.

[4] K. Okazaki, "Body temperature regulation during exercise training," in *Musculoskeletal Disease Associated with Diabetes Mellitus*, pp. 253–268, Springer, Berlin, Germany, 2016.

[5] J.-K. Davis and P. A. Bishop, "Impact of clothing on exercise in the heat," *Sports Medicine*, vol. 43, no. 8, pp. 695–706, 2013.

[6] J. Jiao, *Effects of clothing on running physiology and performance in a hot condition [Ph.D. thesis]*, The Hong Kong Polytechnic University, 2014.

[7] R. J. Dear, T. Akimoto, E. A. Arens et al., "Progress in thermal comfort research over the last twenty years," *Indoor Air*, vol. 23, no. 6, pp. 442–461, 2013.

[8] A. K. Mishra and M. Ramgopal, "Field studies on human thermal comfort—an overview," *Building and Environment*, vol. 64, pp. 94–106, 2013.

[9] R. Wang, H. Du, F. Zhou, D. Deng, and Y. Liu, "An adaptive neural fuzzy network clothing comfort evaluation model and application in digital home," *Multimedia Tools and Applications*, vol. 71, no. 2, pp. 395–410, 2014.

[10] B. R. M. Kingma, L. Schellen, A. J. H. Frijns, and W. D. van Marken Lichtenbelt, "Thermal sensation: a mathematical model based on neurophysiology," *Indoor Air*, vol. 22, no. 3, pp. 253–262, 2012.

[11] C. Huizenga, Z. Hui, and E. Arens, "A model of human physiology and comfort for assessing complex thermal environments," *Building and Environment*, vol. 36, no. 6, pp. 691–699, 2001.

[12] P. Senthilkumar, "Heat and moisture transfer in textiles," in *Man-Made Textiles in India*, p. 43, 2015.

[13] M. Fu, T. Yu, H. Zhang, W. Weng, and H. Yuan, "Heat and moisture transfer through clothing for a person with contact surface," in *Proceedings of the 13th International Conference on Indoor Air Quality and Climate, Indoor Air*, pp. 100–107, Hong Kong, July 2014.

[14] P. S. H. Henry, "The diffusion of moisture and heat through textiles," *Discussions of the Faraday Society*, vol. 3, pp. 243–257, 1948.

[15] B. Farnworth, "Mechanisms of heat flow through clothing insulation," *Textile Research Journal*, vol. 53, no. 12, pp. 717–725, 1983.

[16] Y. Li and B. V. Holcombe, "A two-stage sorption model of the coupled diffusion of moisture and heat in wool fabrics," *Textile Research Journal*, vol. 62, no. 4, pp. 211–217, 1992.

[17] Y. Li and Z. Luo, "An improved mathematical simulation of the coupled diffusion of moisture and heat in wool fabric," *Textile Research Journal*, vol. 69, no. 10, pp. 760–768, 1999.

[18] Y. Lu, G. Song, H. Zeng, L. Zhang, and J. Li, "Characterizing factors affecting the hot liquid penetration performance of fabrics for protective clothing," *Textile Research Journal*, vol. 84, no. 2, pp. 174–186, 2014.

[19] M. Fu, M. Q. Yuan, and W. G. Weng, "Modeling of heat and moisture transfer within firefighter protective clothing with the moisture absorption of thermal radiation," *International Journal of Thermal Sciences*, vol. 96, pp. 201–210, 2015.

[20] N. Sarier and E. Onder, "Organic phase change materials and their textile applications: an overview," *Thermochimica Acta*, vol. 540, pp. 7–60, 2012.

[21] L. Fengzhi, L. Yi, L. Yingxi, and L. Zhongxuan, "Numerical simulation of coupled heat and mass transfer in hygroscopic porous materials considering the influence of atmospheric pressure," *Numerical Heat Transfer, Part B: Fundamentals*, vol. 45, no. 3, pp. 249–262, 2004.

[22] Y. Li and Q. Zhu, "A model of coupled liquid moisture and heat transfer in porous textiles with consideration of gravity," *Numerical Heat Transfer; Part A: Applications*, vol. 43, no. 5, pp. 501–523, 2003.

[23] Y. Cheng, J. Niu, and N. Gao, "Thermal comfort models: a review and numerical investigation," *Building and Environment*, vol. 47, no. 1, pp. 13–22, 2012.

[24] M. Fu, W. Weng, W. Chen, and N. Luo, "Review on modeling heat transfer and thermoregulatory responses in human body," *Journal of Thermal Biology*, vol. 62, pp. 189–200, 2016.

[25] B. Givoni and R. F. Goldman, "Predicting rectal temperature response to work, environment, and clothing," *Journal of Applied Physiology*, vol. 32, no. 6, pp. 812–822, 1972.

[26] A. P. Gagge, A. Fobelets, and L. Berglund, "A standard predictive index of human response to the thermal environment," *ASHRAE Transactions*, vol. 92, pp. 709–731, 1986.

[27] G. Fu, *A transient 3-D mathematical thermal model for the clothed human [Ph.D. thesis]*, Kansas State Unviersity, 1995.

[28] J. A. Stolwijk, *A Mathematical Model of Physiological Temperature Regulation in Man*, National Aeronautics and Space Administration, 1971.

[29] D. Fiala, G. Havenith, P. Bröde, B. Kampmann, and G. Jendritzky, "UTCI-Fiala multi-node model of human heat transfer and temperature regulation," *International Journal of Biometeorology*, vol. 56, no. 3, pp. 429–441, 2012.

[30] Y. Tang, Y. He, H. Shao, and C. Ji, "Assessment of comfortable clothing thermal resistance using a multi-scale human thermoregulatory model," *International Journal of Heat and Mass Transfer*, vol. 98, pp. 568–583, 2016.

[31] W. G. Weng, X. F. Han, and M. Fu, "An extended multi-segmented human bioheat model for high temperature environments," *International Journal of Heat and Mass Transfer*, vol. 75, pp. 504–513, 2014.

[32] L. Yi, M. Aihua, W. Ruomei et al., "P-smart—a virtual system for clothing thermal functional design," *Computer-Aided Design*, vol. 38, no. 7, pp. 726–739, 2006.

[33] Y. Li, "The science of clothing comfort," *Textile Progress*, vol. 31, no. 1-2, pp. 1–135, 2001.

[34] A. S. W. Wong, Y. Li, P. K. W. Yeung, and P. W. H. Lee, "Neural network predictions of human psychological perceptions of clothing sensory comfort," *Textile Research Journal*, vol. 73, no. 1, pp. 31–37, 2003.

[35] A. S. W. Wong, Y. Li, and P. K. W. Yeung, "Predicting clothing sensory comfort with artificial intelligence hybrid models," *Textile Research Journal*, vol. 74, no. 1, pp. 13–19, 2004.

[36] Z. Wang, Y. Li, Y. L. Kowk, and C. Y. Yeung, "Mathematical simulation of the perception of fabric thermal and moisture sensations," *Textile Research Journal*, vol. 72, no. 4, pp. 327–334, 2002.

[37] X. Luo, W. Hou, Y. Li, and Z. Wang, "A fuzzy neural network model for predicting clothing thermal comfort," *Computers and Mathematics with Applications*, vol. 53, no. 12, pp. 1840–1846, 2007.

[38] M. J. Buller, W. J. Tharion, S. N. Cheuvront et al., "Estimation of human core temperature from sequential heart rate observations," *Physiological Measurement*, vol. 34, no. 7, article 781, 2013.

[39] A. Mao, Y. Li, X. Luo, R. Wang, and S. Wang, "A CAD system for multi-style thermal functional design of clothing," *CAD Computer Aided Design*, vol. 40, no. 9, pp. 916–930, 2008.

[40] A. Mao, J. Luo, Y. Li, R. Wang, G. Li, and Y. Guo, "Engineering design of thermal quality clothing on a simulation-based and lifestyle-oriented CAD system," *Engineering with Computers*, vol. 27, no. 4, pp. 405–421, 2011.

[41] W. L. Kenney, J. Wilmore, and D. Costill, *Physiology of Sport and Exercise*, Human Kinetics, 6th edition, 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 4397061, 12 pages
http://dx.doi.org/10.1155/2016/4397061

Research Article

Time-Aware IoE Service Recommendation on Sparse Data

Lianyong Qi,[1,2] Xiaolong Xu,[3] Wanchun Dou,[2] Jiguo Yu,[1] Zhili Zhou,[3] and Xuyun Zhang[4]

[1]*School of Information Science and Engineering, Qufu Normal University, Qufu, China*
[2]*Department of Computer Science and Technology, State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China*
[3]*School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China*
[4]*Department of Electrical and Computer Engineering, University of Auckland, Auckland, New Zealand*

Correspondence should be addressed to Lianyong Qi; lianyongqi@gmail.com

Received 15 September 2016; Accepted 10 November 2016

Academic Editor: Beniamino Di Martino

With the advent of "Internet of Everything" (IoE) age, an excessive number of IoE services are emerging on the web, which places a heavy burden on the service selection decision of target users. In this situation, various recommendation techniques are introduced to alleviate the burden, for example, Collaborative Filtering- (CF-) based recommendation. Generally, CF-based recommendation approaches utilize similar friends or similar services to achieve the recommendation goal. However, due to the sparsity of user feedback, it is possible that a target user has no similar friends and similar services; in this situation, traditional CF-based approaches fail to produce a satisfying recommendation result. Besides, recommendation accuracy would be decreased if time factor is overlooked, as IoE service quality often varies with time. In view of these challenges, a time-aware service recommendation approach named $\text{Ser\_Rec}_{\text{time}}$ is proposed in this paper. Concretely, we first calculate the time-aware user similarity; afterwards, indirect friends of the target user are inferred by Social Balance Theory (e.g., "enemy's enemy is a friend" rule); finally, the services preferred by indirect friends of the target user are recommended to the target user. At last, through a set of experiments deployed on dataset WS-DREAM, we validate the feasibility of our proposal.

1. Introduction

With the wide adoption of various smart devices and connection technologies, human society is gradually transforming into an "Internet of Everything" (IoE) one [1–5]. In the age of IoE, the links among users, devices, or other things could be built easily, which significantly improves people's quality of life and also brings many challenging open problems that need to be addressed [6–11].

With the advent of IoE age, an excessive number of IoE services with different functionalities or qualities are emerging on the web, which places a heavy burden on the service selection decision of target users [12–14]. In this situation, various recommendation techniques are put forward to help alleviate the service selection burden, for example, Collaborative Filtering- (CF-) based recommendation [15–19]. Generally, CF-based recommendation approaches (including user-based CF, item-based CF, and hybrid CF) utilize the similar friends or similar services of target users to achieve the recommendation goal. However, in certain situations, the available user-service rating data generated from historical service invocations is really sparse [20]. Therefore, it is probable that a target user cannot find his/her similar friends and similar services of target services (here, target services mean the services preferred by target users). In this situation, traditional CF-based recommendation approaches cannot return a satisfying recommendation result, which brings a great challenge for the robustness of service recommendation approaches. Besides, the quality of an IoE service often varies with time, due to the unstable network environment [21]. For example, the response time of a ticket-order service often becomes larger when Christmas day is approaching. Therefore, recommendation accuracy would be decreased if the time factor is not taken into consideration.

In view of the above two challenges, a novel time-aware service recommendation approach, that is, $\text{Ser\_Rec}_{\text{time}}$, is put forward in this paper, to make robust and accurate service recommendation for target users when the historical

user-service rating data is sparse. Concretely, in $\text{Ser\_Rec}_{\text{time}}$, we first calculate the time-aware user similarity; afterwards, we look for the enemies (antonym of "friend") of the target user and further determine the "indirect friends" of target user by Social Balance Theory [22] (e.g., "enemy's enemy is a friend" rule); finally, the services preferred by indirect friends of target user are recommended to the target user.

The contributions of our paper are threefold.

(1) Time factor is considered in user similarity calculation to adapt to the dynamic quality variation of IoE services, which makes the subsequent recommendation result more objective and accurate.

(2) Social Balance Theory is introduced for service recommendation on sparse data so that the recommendation robustness is improved.

(3) A wide range of experiments are designed and deployed on a real web service quality set WS-DREAM [23], so as to further validate the feasibility of our proposal.

The remainder of our paper is organized as follows. In Section 2, we first formalize the CF-based service recommendation problem and afterwards demonstrate the motivation of our paper. In Section 3, a novel time-aware service recommendation approach named $\text{Ser\_Rec}_{\text{time}}$ is brought forth. A set of experiments are deployed in Section 4 and evaluations are presented in Section 5. Finally, in Section 6, we summarize the paper and point out our future research directions.

2. Formalization and Motivation

In this section, we first formalize the CF-based service recommendation problem. And afterwards, an example is presented to demonstrate the motivation of our paper intuitively.

2.1. Formal Specification. Generally, the CF-based service recommendation problem could be specified with a four-tuple CF-Ser-Rec $(U, \text{WS}, \rightarrow, \text{user}_{\text{target}})$, where

(1) $U = \{\text{user}_1, \ldots, \text{user}_m\}$ denotes the user set in user-service invocation network and m is the number of users;

(2) $\text{WS} = \{\text{ws}_1, \ldots, \text{ws}_n\}$ denotes the web service set in user-service invocation network and n is the number of web services;

(3) $\rightarrow = \{(\text{user}_i \xrightarrow[T_{i-j}]{R_{i-j}} \text{ws}_j) \mid 1 \le i \le m,\ 1 \le j \le n\}$ denotes the historical invocation record set, that is, $\text{user}_i \xrightarrow[T_{i-j}]{R_{i-j}} \text{ws}_j$, which means that user_i invoked ws_j in the past and the rating of ws_j by user_i is R_{i-j}. Here, for simplicity, the well-known $\{1^*, 2^*, 3^*, 4^*, 5^*\}$ rating system is adopted to depict R_{i-j}. Also, each invocation record owns a timestamp T_{i-j} that indicates the service invocation time (here, we assume the timestamp format is "yyyy.mm.dd");

(4) $\text{user}_{\text{target}}$ denotes the target user that requires service recommendation and $\text{user}_{\text{target}} \in U$ holds.

With the above formalization, CF-based service recommendation problem could be specified as follows: according to the historical invocation record set "$\rightarrow$" between users (in U) and web services (in WS), find out the services that were never invoked but may be preferred by target user $\text{user}_{\text{target}}$ and recommend them to $\text{user}_{\text{target}}$.

2.2. Motivation. In this subsection, the paper motivation is clarified more intuitively with the example shown in Figure 1. In Figure 1, there are three users {Tom, Alice, Bob} (Tom is the target user) in set U and six web services $\{\text{ws}_1, \text{ws}_2, \text{ws}_3, \text{ws}_4, \text{ws}_5, \text{ws}_6\}$ in set WS; historical user-service invocation record set $\rightarrow$ (including rating data and timestamp) is also presented in Figure 1.

Then according to the Adjusted Cosine Similarity [24] (i.e., $\text{ACS} \in [-1, 1]$; here, the reason that we utilize ACS for similarity calculation is that user-service rating data is often discrete and the rating scales of different users often vary), we can calculate the similarity between target user Tom and other two users (i.e., Alice and Bob). Concretely, $\text{Sim}(\text{Tom}, \text{Alice}) = -0.2747$ and $\text{Sim}(\text{Tom}, \text{Bob}) = \text{Null}$ (as no services were rated by both Tom and Bob). Therefore, we can conclude that target user Tom has no similar friends, so according to the traditional user-based CF recommendation approaches, no qualified services are recommended to Tom.

Likewise, the similarities between target services (i.e., ws_1 and ws_2) and other services (i.e., ws_3, ws_4, ws_5, and ws_6) could also be obtained. Concretely, $\text{Sim}(\text{ws}_1, \text{ws}_3) = \text{Sim}(\text{ws}_1, \text{ws}_4) = \text{Sim}(\text{ws}_2, \text{ws}_3) = \text{Sim}(\text{ws}_2, \text{ws}_4) = -1$ and $\text{Sim}(\text{ws}_1, \text{ws}_5) = \text{Sim}(\text{ws}_1, \text{ws}_6) = \text{Sim}(\text{ws}_2, \text{ws}_5) = \text{Sim}(\text{ws}_2, \text{ws}_6) = \text{Null}$ (as no users invoked any pair of services above simultaneously). Therefore, a conclusion could be drawn that target user Tom's preferred services (i.e., ws_1 and ws_2) have no similar services, so according to the traditional item-based CF recommendation approaches, no qualified services are recommended to Tom.

Therefore, in this situation, traditional CF-based service recommendation approaches (e.g., user-based CF, item-based CF, or hybrid CF) cannot make accurate service recommendation for target user. Besides, as Figure 1 shows, the timestamps of various invocation records are often different, so the service recommendation accuracy and fairness would be decreased if we overlook the time factor in recommendation process. In view of the above two challenges, a novel service recommendation approach named $\text{Ser\_Rec}_{\text{time}}$ is put forward in Section 3. $\text{Ser\_Rec}_{\text{time}}$ cannot only find out the indirect friends of a target user so as to improve the recommendation robustness in sparse-data environment, but also consider time factor in recommendation so as to ensure the fairness and accuracy of recommendation results.

3. Time-Aware Service Recommendation Approach: $\text{Ser\_Rec}_{\text{time}}$

In this section, a novel time-aware recommendation approach, that is, $\text{Ser\_Rec}_{\text{time}}$, is introduced to deal with the

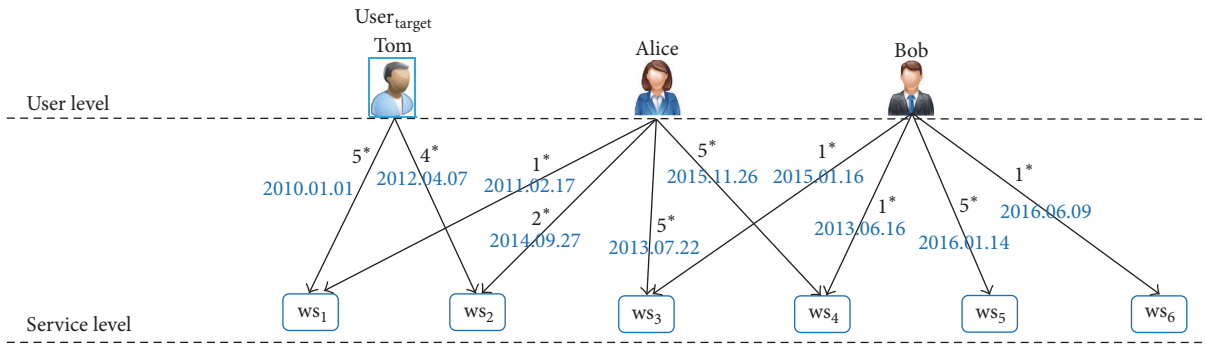


Figure 1: Time-aware service recommendation when target user has no similar friends and similar services.

service recommendation problem on sparse data. The main idea of our proposal is as follows: first, we calculate the time-aware similarity between target user and other users based on the historical user-service rating records; second, according to the derived user similarity and Social Balance Theory, we infer the indirect friends of target user; third, the services preferred by indirect friends of target user are recommended to the target user. Concretely, the three steps of time-aware service recommendation approach (Ser_Rec$_{time}$) are as follows.

Step 1 (time-aware user similarity calculation). Calculate the similarity Sim(user$_{target}$, user$_i$) between user$_{target}$ and other users user$_i$ $\in U$. If Sim(user$_{target}$, user$_i$) $\leq P$ (P is similarity threshold), then user$_i$ is considered as an enemy of user$_{target}$.

Step 2 (determining indirect friends of target user). According to the enemies (derived in Step 1) of target user and Social Balance Theory, determine target user's indirect friends.

Step 3 (service recommendation). Select the services preferred by the indirect friends (derived in Step 2) of target user and recommend them to the target user.

The three steps are explained as follows:

Step 1 (time-aware user similarity calculation). In this step, we calculate the similarity between target user user$_{target}$ and other users user$_i$ (user$_i \in U$), that is, Sim(user$_{target}$, user$_i$). As user-service rating data is often discrete and rating scores of a service by different users are often varied, Adjusted Cosine Similarity is recruited here for user similarity calculation. Concretely, Sim(user$_{target}$, user$_i$) could be obtained based on

$$\mathrm{Sim}\left(\mathrm{user}_{target}, \mathrm{user}_i\right) = \frac{\sum_{\mathrm{ws}_j \in I}\left(R_{\mathrm{target}-j} - \overline{R_{\mathrm{target}}}\right) * \left(R_{i-j} - \overline{R_i}\right)}{\sqrt{\sum_{\mathrm{ws}_j \in I_{target}}\left(R_{\mathrm{target}-j} - \overline{R_{\mathrm{target}}}\right)^2} * \sqrt{\sum_{\mathrm{ws}_j \in I_i}\left(R_{i-j} - \overline{R_i}\right)^2}}. \quad (1)$$

Here, set I denotes the common services that were rated by both user$_{target}$ and user$_i$; sets I_{target} and I_i denote the service set rated by user$_{target}$ and user$_i$, respectively; $R_{target-j}$ and R_{i-j} denote ratings of service ws$_j$ by user$_{target}$ and user$_i$, respectively; while $\overline{R_{target}}$ and $\overline{R_i}$ represent user$_{target}$'s and user$_i$'s average rating values. Then according to (1), we can calculate the similarity between target user and any other user.

Next, we improve the user similarity formula in (1) by introducing four kinds of weight coefficients.

(i) *Weight for Service Intersection Size*. As formula (1) indicates, I_{target} and I_i denote the service set rated by user$_{target}$ and user$_i$, respectively. Generally, for user$_{target}$ and user$_i$, the larger their service intersection (i.e., set I in formula (1)) is, the more convincing their similarity is. So in order to depict this correlation, weight $W_{\text{ser-intersection}}$ (in formula (2)) is assigned to user similarity Sim(user$_{target}$, user$_i$):

$$W_{\text{ser-intersection}} = \frac{1}{2} * \left(\frac{I_{target} \cap I_i}{I_{target}} + \frac{I_{target} \cap I_i}{I_i}\right). \quad (2)$$

(ii) *Weight for Invocation Time*. Due to the dynamic and unstable network environment, the IoE service quality is often varied with time; therefore, two neighboring service invocations with close invocation time often contribute more to the user similarity. In view of this observation, similar to work [4], weight W_{time} in formula (3) is assigned to user similarity Sim(user$_{target}$, user$_i$). In (3), α ($\alpha \geq 0$) is a parameter, while $T_{target-j}$ and T_{i-j} represent the invocation time of service ws$_j$ by user$_{target}$ and user$_i$, respectively

$$W_{time} = e^{-\alpha\left|T_{target-j} - T_{i-j}\right|}. \quad (3)$$

(iii) *Weight for Invocation Load*. Service invocation time cannot reflect all the time-related information in similarity calculation. For example, user$_1$ and user$_2$ invoked the same ticket-booking service on 22-12-2015 and 24-12-2015, respectively. Although their service invocation time is close, the user experienced service quality may vary significantly as heavy service load is inevitable when Christmas day is approaching. So in order to depict the effect of service load on user similarity, weight W_{load} in formula (4) is assigned to user similarity Sim(user$_{target}$, user$_i$). In (4), Load$_{target-j}$ and Load$_{i-j}$ denote the service

loads when $\text{user}_{\text{target}}$ and user_i invoked service ws_j, respectively.

$$W_{\text{load}} = 1 - \frac{\left|\text{Load}_{\text{target}-j} - \text{Load}_{i-j}\right|}{\max\left\{\text{Load}_{\text{target}-j}, \text{Load}_{i-j}\right\}}. \tag{4}$$

(iv) *Weight for Service Version*. In the life cycle of a web service, service provider may publish a series of service versions with updated service qualities. Therefore, if two users invoked an identical web service that belongs to different versions, it would not make much sense to compare their experienced service qualities. In view of this observation, weight W_{version} is suggested as follows:

$$W_{\text{version}} = \begin{cases} 1 & \text{user}_{\text{target}}, \text{user}_i \text{ invoked same service of same version} \\ 0 & \text{else.} \end{cases} \tag{5}$$

With the above analyses, we can update the user similarity $\text{Sim}(\text{user}_{\text{target}}, \text{user}_i)$ in (1) to be time-aware user similarity $\text{Sim}_{\text{time}}(\text{user}_{\text{target}}, \text{user}_i)$ in (6). Then based on (6), we can calculate the time-aware similarity between target user and any other user:

$$\text{Sim}_{\text{time}}\left(\text{user}_{\text{target}}, \text{user}_i\right) = W_{\text{ser-intersection}} * \frac{\sum_{\text{ws}_j \in I} W_{\text{time}} * W_{\text{load}} * W_{\text{version}} * \left(R_{\text{target}-j} - \overline{R_{\text{target}}}\right) * \left(R_{i-j} - \overline{R_i}\right)}{\sqrt{\sum_{\text{ws}_j \in I_{\text{target}}} \left(R_{\text{target}-j} - \overline{R_{\text{target}}}\right)^2} * \sqrt{\sum_{\text{ws}_j \in I_i} \left(R_{i-j} - \overline{R_i}\right)^2}}. \tag{6}$$

Furthermore, according to the derived user similarity, the enemy set of target user, that is, $\text{Enemy\_set}(\text{user}_{\text{target}})$, could be obtained based on (7). Here, parameter P ($-1 \leq P \leq -0.5$) is a predefined user similarity threshold for enemy relationship:

$$\text{user}_i \begin{cases} \in \text{Enemy\_set}\left(\text{user}_{\text{target}}\right) & \text{if } \text{Sim}_{\text{time}}\left(\text{user}_{\text{target}}, \text{user}_i\right) \leq P \\ \notin \text{Enemy\_set}\left(\text{user}_{\text{target}}\right) & \text{else.} \end{cases} \tag{7}$$

Step 2 (determining indirect friends of target user). In Step 1, we have obtained the enemy set of target user, that is, $\text{Enemy\_set}(\text{user}_{\text{target}})$. Next, in this step, we will introduce how to get the indirect friends of target user, that is, $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$, based on the obtained $\text{Enemy\_set}(\text{user}_{\text{target}})$ and Social Balance Theory. First of all, we introduce Social Balance Theory briefly.

Social Balance Theory [22] was first put forward by psychologist F. Heider in 1958. The theory investigates the stable social relationships among involved three parties (i.e., P, O, and X in Figure 2). Concretely, according to Figure 2, we introduce the four stable social relationships, respectively, in a more intuitive manner.

(a) *Friend's Friend Is a Friend*. If X is a friend of O and O is a friend of P, then we can infer that X is probably an indirect friend of P (see Figure 2(a)).

(b) *Enemy's Enemy Is a Friend*. If X is an enemy of O and O is an enemy of P, then we can infer that X is probably an indirect friend of P (see Figure 2(b)).

(c) *Friend's Enemy Is an Enemy*. If X is an enemy of O while O is a friend of P, then we can infer that X is probably an indirect enemy of P (see Figure 2(c)).

(d) *Enemy's Friend Is an Enemy*. If X is a friend of O while O is an enemy of P, then we can infer that X is probably an indirect enemy of P (see Figure 2(d)).

Next, with the above four rules in Social Balance Theory, we introduce how to obtain the indirect friend set of target user, that is, $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$, based on the derived $\text{Enemy\_set}(\text{user}_{\text{target}})$ in Step 1.

Concretely, for each $\text{user}_i \in \text{Enemy\_set}(\text{user}_{\text{target}})$, we first calculate his/her similarities with other users based on (6); afterwards, we determine user_i's enemies user_k (i.e., $\text{user}_k \in \text{Enemy\_set}(\text{user}_i)$) based on (7) and user_i's friends user_g (i.e., $\text{user}_g \in \text{Friend\_set}(\text{user}_i)$) based on (8). In (8), parameter $-P$ denotes the user similarity threshold for friend relationship. To ease the understanding of readers, the relationships among $\text{user}_{\text{target}}$, user_i, user_k, and user_g are presented in Figure 3.

$$\text{user}_g \begin{cases} \in \text{Friend\_set}\left(\text{user}_i\right) & \text{if } \text{Sim}_{\text{time}}\left(\text{user}_i, \text{user}_g\right) \geq -P \\ \notin \text{Friend\_set}\left(\text{user}_i\right) & \text{else.} \end{cases} \tag{8}$$

Then according to "enemy's enemy is a friend" rule in Figure 2(b), we can infer that user_k is probably an indirect friend of $\text{user}_{\text{target}}$, and the probability could be calculated based on (9). If $\text{Probability}_{\text{friend}}(\text{user}_{\text{target}}, \text{user}_k) \geq -P$, then user_k is regarded as a qualified indirect friend of target user and put into set $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$ (see Figure 3(b)). Likewise, according to "enemy's friend is an enemy" rule in Figure 2(d), it can be inferred that user_g is probably an indirect enemy of $\text{user}_{\text{target}}$, and the probability can be obtained based on (10). If $\text{Probability}_{\text{enemy}}(\text{user}_{\text{target}}, \text{user}_g) \geq -P$, then user_g is regarded as a qualified indirect enemy of target user and put into set $\text{Enemy\_set}(\text{user}_{\text{target}})$ (see Figure 3(d))

$$\begin{aligned} &\text{Probability}_{\text{friend}}\left(\text{user}_{\text{target}}, \text{user}_k\right) \\ &= \text{Sim}_{\text{time}}\left(\text{user}_{\text{target}}, \text{user}_i\right) \\ &* \text{Sim}_{\text{time}}\left(\text{user}_i, \text{user}_k\right), \end{aligned} \tag{9}$$

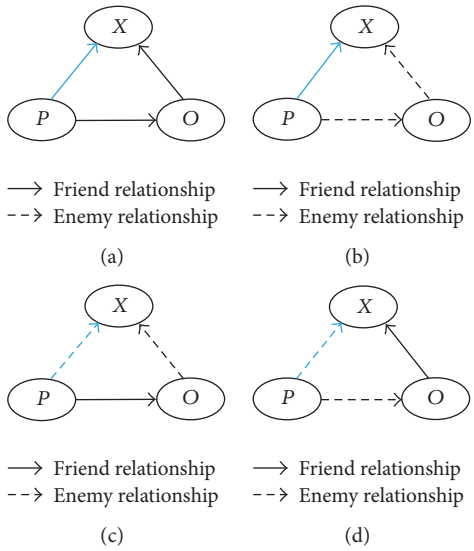


Figure 2: Four stable relationships among P, X, and O according to Social Balance Theory (the blue arrows between P and X denote the inferred relationships).

$$\begin{aligned}&\text{Probability}_{\text{enemy}}\left(\text{user}_{\text{target}}, \text{user}_g\right)\\&= \left|\text{Sim}_{\text{time}}\left(\text{user}_{\text{target}}, \text{user}_i\right)\right.\\&\left.* \text{Sim}_{\text{time}}\left(\text{user}_i, \text{user}_g\right)\right|.\end{aligned} \tag{10}$$

Next, for each $\text{user}_x \in \text{Indirect\_friend\_set}(\text{user}_{\text{target}})$, we calculate his/her similarity with other users ($\in (U\text{-user}_{\text{target}}\text{-Indirect\_friend\_set}(\text{user}_{\text{target}})\text{-Enemy\_set}(\text{user}_{\text{target}}))$) based on (6) and further determine user_x's enemies user_y based on (7) and user_x's friends user_z based on (8). Then according to "friend's enemy is an enemy" rule in Figure 2(c), user_y is regarded as an indirect enemy of target user and put into Enemy_set ($\text{user}_{\text{target}}$) (see Figure 3(c)) if the probability in (11) is larger than $-P$. Likewise, according to "friend's friend is a friend" rule in Figure 2(a), user_z is considered as an indirect friend of target user and put into Indirect_friend_set ($\text{user}_{\text{target}}$) (see Figure 3(a)) if the probability in (12) is larger than $-P$.

$$\begin{aligned}&\text{Probability}_{\text{enemy}}\left(\text{user}_{\text{target}}, \text{user}_y\right)\\&= \left|\text{Probability}_{\text{friend}}\left(\text{user}_{\text{target}}, \text{user}_x\right)\right.\\&\left.* \text{Sim}_{\text{time}}\left(\text{user}_x, \text{user}_y\right)\right|,\end{aligned} \tag{11}$$

$$\begin{aligned}&\text{Probability}_{\text{friend}}\left(\text{user}_{\text{target}}, \text{user}_z\right)\\&= \text{Probability}_{\text{friend}}\left(\text{user}_{\text{target}}, \text{user}_x\right)\\&* \text{Sim}_{\text{time}}\left(\text{user}_x, \text{user}_z\right).\end{aligned} \tag{12}$$

Repeat (a)–(d) process in Figure 3 until set $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$ stays stable. Then we obtain the indirect friends of target user, that is, $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$.

Step 3 (service recommendation). In Step 2, we have obtained target user's indirect friend set $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$. Next, in this step, we select the services preferred (i.e., with 4^* or 5^* rating) by indirect friends of target user and recommend them to the target user. More formally, for each element $\text{user}_x \in \text{Indirect\_friend\_set}(\text{user}_{\text{target}})$, if his/her rating over web service ws_j ($1 \le j \le n$), that is, $R_{x-j} \in \{4^*, 5^*\}$, holds, then ws_j is put into the recommended service set, that is, Rec_Ser_Set. Finally, all the web services in set Rec_Ser_Set are recommended to $\text{user}_{\text{target}}$.

With above Step 1–Step 3 of our proposed $\text{Ser\_Rec}_{\text{time}}$ approach, a set of IoE services (in set Rec_Ser_Set) are recommended to the target user. More formally, the pseudocode of $\text{Ser\_Rec}_{\text{time}}(U, \text{WS}, \rightarrow, \text{user}_{\text{target}})$ is presented in Algorithm 1 (please note that algorithm $\text{Ser\_Rec}_{\text{time}}(U, \text{WS}, \rightarrow, \text{user}_{\text{target}})$ is abbreviated as $\text{Ser\_Rec}_{\text{time}}$ in the whole paper).

4. Experiment

In this section, a set of experiments are designed and tested to validate the feasibility of our proposed $\text{Ser\_Rec}_{\text{time}}$ approach, in terms of recommendation accuracy, recall, and efficiency.

4.1. Experiment Dataset and Deployment. The experiment is based on a real service quality dataset WS-DREAM [23]. WS-DREAM consists of 4532 IoE services on the web, and 142 distributed users from Planet-Lab are employed for evaluating the real quality (e.g., *response time* and *throughput*) of services in 64 time intervals (time interval = 15 minutes).

Our paper focuses on the user-service rating-based recommendation, while available user-service rating data is really rare on the web; therefore, in the experiment, we need to transform the service quality data in WS-DREAM into corresponding user-service rating data (essentially, objective service quality data and subjective user-service rating data both reflect the service running quality; therefore, we argue that the transformation from former data to latter data makes sense for the service recommendation simulation here). Concretely, the transformation process is as follows: we determine the minimal and maximal quality values (denoted by min and max, resp.) of a service observed by an identical user, and afterwards we divide the range [min, max] into five subranges in an arithmetic progression manner, each corresponding to a rating value. For example, if the minimal and maximal throughput values of a service ws_j observed by user_i are 10 kbps and 60 kbps, respectively, then we can get five subranges after division, that is, [10, 20) kbps, [20, 30) kbps, [30, 40) kbps, [40, 50) kbps, and [50, 60] kbps. Furthermore, if the throughput value of ws_j observed by user_i is 44 kbps in WS-DREAM, then the transformed user-service rating, that is, $R_{i-j} = 4^*$, holds. For each service invoked by a user, we randomly select a time interval from all the 64 intervals and

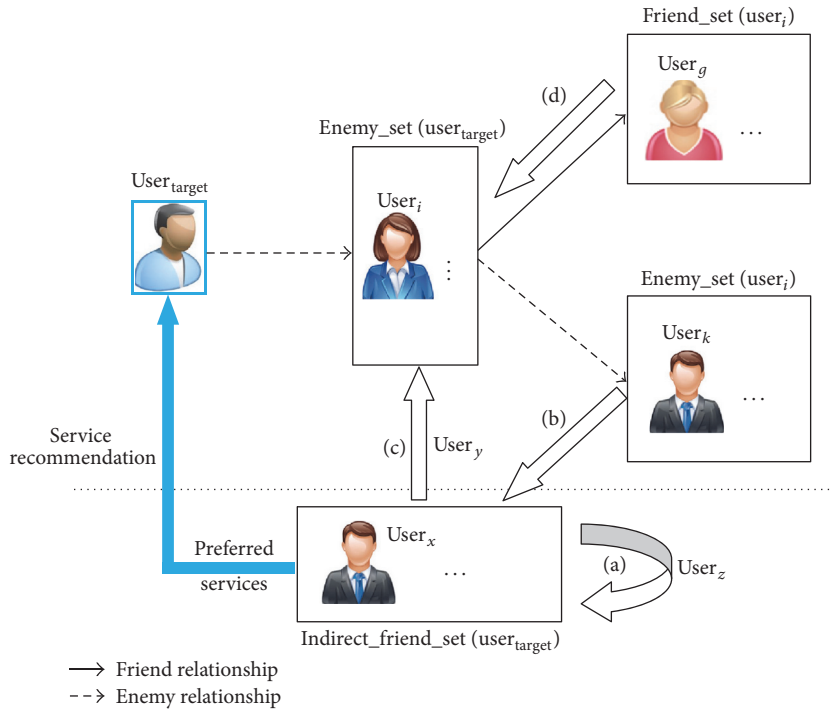


Figure 3: Relationships among user_{target}, user_i, user_k, user_g, user_x, user_y, and user_z.

transform the service quality corresponding to the selected time interval into a user-service rating with a timestamp (i.e., the selected time interval). In the experiment, only a quality dimension, that is, *throughput*, is considered.

Besides, our paper only discusses the time-aware service recommendation in sparse-data environment where target user has no similar friends and similar services. Therefore, we select appropriate data from WS-DREAM to simulate the above sparse recommendation situations. Furthermore, for each recruited target user, his/her 80% ratings are randomly selected for training while the remaining 20% ratings for testing.

Also, we compare our proposal with related works, for example, WSRec [25], MCCP [26], and SBT-SR [27], in terms of recommendation accuracy and recall. Concretely, accuracy and recall are measured by Mean Absolute Error (i.e., MAE; the smaller the better) in (13) and Recall (the larger the better) in (14). In (13), Rec_Ser_Set denotes the recommended service set for target user; $|X|$ denotes the size of set X; $R_{target-x}$ and $R^{\#}_{target-x}$ represent target user's real and predicted ratings over service ws_x, respectively. While in (14), symbols Preferred_Ser_Set and Rec_Ser_Set denote the service set really preferred (with 4^* or 5^* rating) by target user and the service set recommended to target user, respectively.

$$\text{MAE} = \sum_{ws_x \in \text{Rec\_Ser\_Set}} \frac{\left|R_{target-x} - R^{\#}_{target-x}\right|}{|\text{Rec\_Ser\_Set}|}, \quad (13)$$

$$\text{Recall} = \frac{|\text{Preferred\_Ser\_Set} \cap \text{Rec\_Ser\_Set}|}{|\text{Preferred\_Ser\_Set}|}. \quad (14)$$

The experiments were conducted on a Dell laptop with 2.80 GHz processors and 2.0 GB RAM. The software configuration is Windows XP + JAVA 1.5. Each experiment was carried out 10 times and the average results were adopted finally.

4.2. Experiment Results and Analyses. Concretely, four profiles are designed, tested, and compared. Here, m and n denote the number of users and number of services in the historical user-service invocation network.

(Profile 1) Recommendation Accuracy Comparison. In this profile, we compare the recommendation accuracy of Ser_Rec_{time} with the other three approaches, that is, WSRec, MCCP, and SBT-SR. Concretely, parameters $w_u = w_i = 0.5$ hold in WSRec; $\alpha = 0.8$ holds in MCCP; user similarity threshold $P = -0.5$ holds in both SBT-SR and Ser_Rec_{time}. Besides, in Ser_Rec_{time}, $\alpha = 0.09$ holds in (3) and $W_{\text{load}} = W_{\text{version}} = 1$ holds (in WS-DREAM, a service was invoked by a user 64 times within successive 16 hours; therefore, we assume that the service load and version do not vary much). Finally, m is varied from 20 to 100 in all the four approaches.

The experiment result is presented in Figure 4. As Figure 4 shows, the recommendation accuracy of WSRec is low, as it only adopts the average ratings of target user and target services without considering the valuable user-service relationships hidden in historical service invocation records. The accuracy of MCCP is improved by considering the user-service relationships; however, it is not very suitable for service recommendation when target user has no similar friends and similar services, as MCCP is essentially a kind

Inputs:
(1) $U = \{\text{user}_1, \ldots, \text{user}_m\}$: a set of users in user-service invocation & rating network;
(2) $\text{WS} = \{\text{ws}_1, \ldots, \text{ws}_n\}$: a set of web services in user-service invocation & rating network;
(3) $\rightarrow = \{(\text{user}_i \xrightarrow[T_{i-j}]{R_{i-j}} \text{ws}_j) \mid 1 \le i \le m,\ 1 \le j \le n\}$: a set of historical user-service rating records;
(4) $\text{user}_{\text{target}}$: target user who requires service recommendation.
Output: Rec_Ser_Set: service set recommended to $\text{user}_{\text{target}}$
(1) Set user similarity threshold P $(-1 \le P \le -0.5)$
(2) **for** each $\text{user}_i \in U$ **do** // Step 1
(3) calculate $\text{Sim}_{\text{time}}(\text{user}_{\text{target}}, \text{user}_i)$ based on (1)–(6)
(4) **if** $\text{Sim}_{\text{time}}(\text{user}_{\text{target}}, \text{user}_i) \le P$
(5) **then** put user_i into set $\text{Enemy\_set}(\text{user}_{\text{target}})$
(6) **end if**
(7) **end for**
(8) **for** each $\text{user}_i \in \text{Enemy\_set}(\text{user}_{\text{target}})$ **do** // Step 2
(9) determine user_i's friend user_g based on (6) and (8)
(10) **if** probability in (10) is larger than $-P$
(11) **then** put user_g into $\text{Enemy\_set}(\text{user}_{\text{target}})$
(12) **end if**
(13) determine user_i's enemy user_k based on (6) and (7)
(14) **if** probability in (9) is larger than $-P$
(15) **then** put user_k into $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$
(16) **end if**
(17) **end for**
(18) **for** each $\text{user}_x \in \text{Indirect\_friend\_set}(\text{user}_{\text{target}})$ **do**
(19) determine user_x's enemy user_y based on (6) and (7)
(20) **if** probability in (11) is larger than $-P$
(21) **then** put user_y into $\text{Enemy\_set}(\text{user}_{\text{target}})$
(22) **end if**
(23) determine user_x's friend user_z based on (6) and (8)
(24) **if** probability in (12) is larger than $-P$
(25) **then** put user_z into $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$
(26) **end if**
(27) **end for**
(28) repeat Line (8)–Line (27) until $\text{Indirect\_friend\_set}(\text{user}_{\text{target}})$ stays stable
(29) $\text{Rec\_Ser\_Set} = \emptyset$ // Step 3
(30) **for** each $\text{user}_x \in \text{Indirect\_friend\_set}(\text{user}_{\text{target}})$ **do**
(31) **for** each $\text{ws}_j \in \text{WS}$ **do**
(32) **if** $R_{x-j} \in \{4^*, 5^*\}$
(33) **then** put ws_j into set Rec_Ser_Set
(34) **end if**
(35) **end for**
(36) **end for**
(37) **return** Rec_Ser_Set

Algorithm 1: $\text{Ser\_Rec}_{\text{time}}$ $(U, \text{WS}, \rightarrow, \text{user}_{\text{target}})$.

of similar user-based service recommendation approach. The recommendation accuracy of SBT-SR is high as "enemy's enemy is a friend" rule of Social Balance Theory is recruited to find the "indirect friends" of target user. Furthermore, our proposed $\text{Ser\_Rec}_{\text{time}}$ outperforms SBT-SR as $\text{Ser\_Rec}_{\text{time}}$ not only considers Social Balance Theory, but also takes time factor into consideration for looking for the really similar "indirect friends" of target user. Besides, as shown in Figure 4, recommendation accuracy values of MCCP, SBT-SR, and $\text{Ser\_Rec}_{\text{time}}$ all increase with the growth of m approximately; this is because more valuable user-service relationship information would be discovered and recruited for service recommendation when there are many users as well as their invocation records.

(Profile 2) Recommendation Recall Comparison. In this profile, we compare the recommendation recall values of four approaches. The parameter settings are the same as those in profile 1. Concrete experiment results are shown in Figure 5.

As Figure 5 shows, the recall of WSRec is low as the "average" idea adopted in WSRec often leads to a low recommendation hit rate. The recommendation recall values of the remaining three approaches all increase with the growth of m; this is because when there are many available historical users,

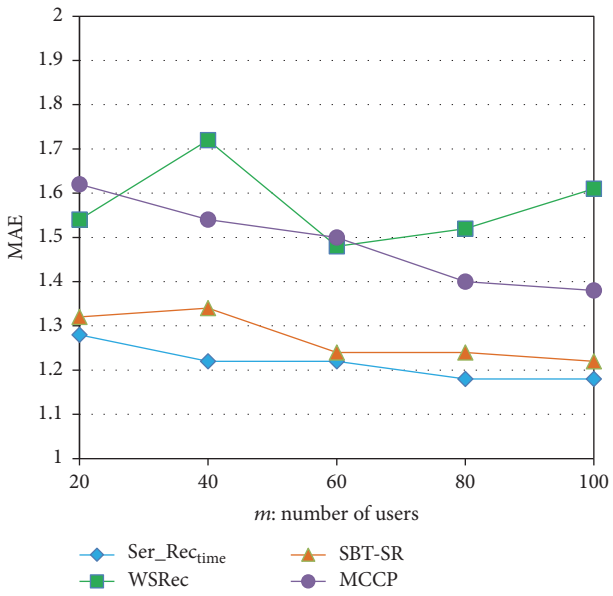


Figure 4: Recommendation accuracy comparison with respect to m.

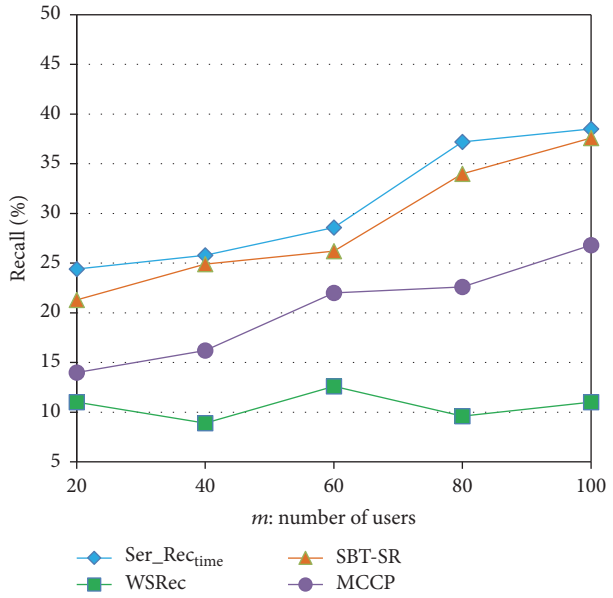


Figure 5: Recommendation recall comparison with respect to m.

more useful user-service relationships could be mined and recruited in service recommendation; hence, more qualified recommendation results are returned finally. However, the recall of MCCP is still not high as few really similar friends of target user could be found in the recommendation situations that we discuss in this paper (i.e., the sparse recommendation situations when target user has no similar friends and similar services).

While both SBT-SR and Ser_Rec$_{time}$ approaches achieve good performances in recommendation recall, as Social Balance Theory is recruited to find out the indirect friends of target user even if the target user has no similar friends

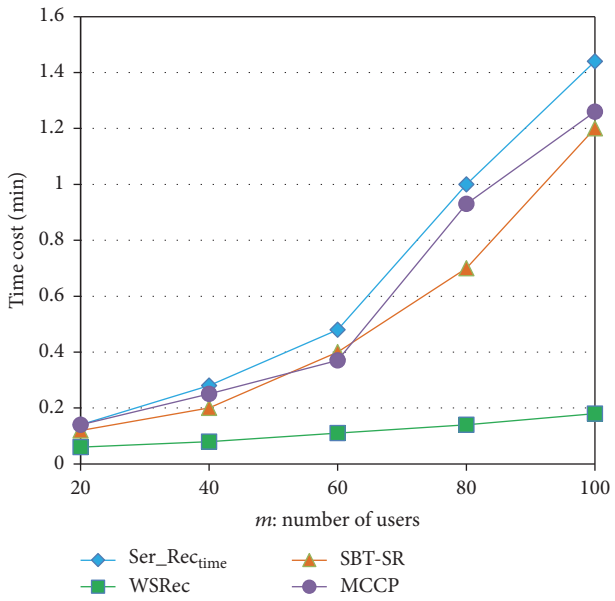


Figure 6: Time cost comparison with respect to m.

and similar services. Furthermore, our proposed Ser_Rec$_{time}$ approach often outperforms SBT-SR in recall. This is because only "enemy's enemy is a friend" rule of Social Balance Theory is recruited in SBT-SR, while in Ser_Rec$_{time}$, all four rules in Figure 2 are considered, which improves the recommendation hit rate to some extent.

(Profile 3) Execution Efficiency Comparison with respect to m. In this profile, we test the execution efficiency of four approaches with respect to the number of users, i.e., m. Here, m is varied from 20 to 100; the number of services, that is, $n = 1000$, holds; besides, $P = -0.5$, $\alpha = 0.09$, and $W_{load} = W_{version} = 1$ hold. The experiment result is shown in Figure 6.

As Figure 6 shows, time cost of WSRec is the best, as WSRec only adopts the average ratings of target user and target services, without complex computation. The time costs of the remaining three approaches all increase with the growth of m quickly, as more similarity computation cost is required to determine the similar friends or dissimilar enemies of a user when the number of users increases. Furthermore, Ser_Rec$_{time}$ often requires more computation time as multiple iteration processes are probable for finding all the indirect friends of target user. However, as Figure 6 indicates, the execution efficiency of Ser_Rec$_{time}$ is often acceptable (at "minute" level).

(Profile 4) Execution Efficiency Comparison with respect to n. In this profile, we test the execution efficiency of four approaches with respect to the number of services, that is, n. Here, n is varied from 200 to 1000; the number of users, that is, $m = 100$, holds; besides, $P = -0.5$, $\alpha = 0.09$, and $W_{load} = W_{version} = 1$ hold. The concrete experiment result is shown in Figure 7.

As Figure 7 shows, similar to profile 3, the time cost of WSRec is the best due to the adopted average idea. Besides,

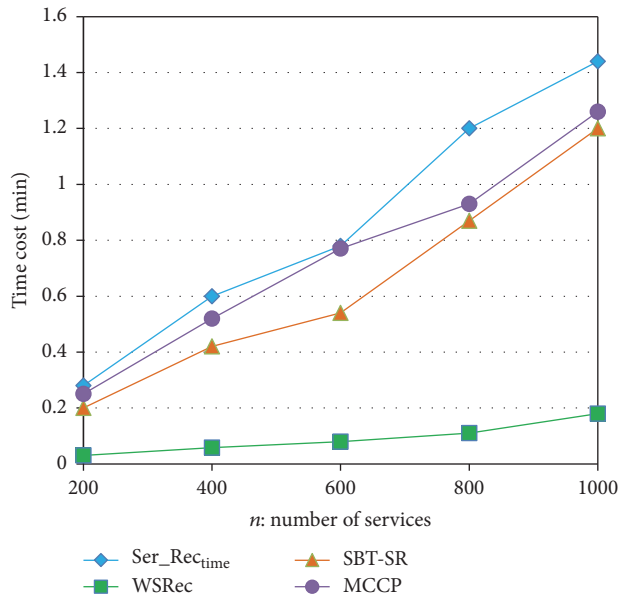


Figure 7: Time cost comparison with respect to n.

the time costs of the remaining three approaches all increase with the growth of m approximately linearly, as each service is considered at most once in each user similarity calculation process. However, as can be seen from Figure 7, the service recommendation process of Ser_Rec$_{time}$ approach could be generally finished in polynomial time.

5. Evaluations

In this section, we first analyze the time complexity of our proposed Ser_Rec$_{time}$ approach. Afterwards, related works and comparison analyses are presented to further clarify the advantages and application scope of our proposal. Finally, we point out our future research directions.

5.1. Complexity Analyses. Suppose there are m users and n services in the historical user-service invocation network.

Step 1. According to (1)–(6), the time-aware similarity between target user and any other user could be calculated, whose time complexity is $O(n)$. Afterwards, a user is judged to be an qualified enemy of target user or not based on (7), whose time complexity is $O(1)$. As there are totally m users in set U, the time complexity of this step is $O(m * (n + 1)) = O(m * n)$.

Step 2. In this step, we utilize the four rules (see Figure 3) in Social Balance Theory iteratively, so as to find all the indirect friends of target user. And, in the worst case, the similarity between any two users in set U needs to be calculated. As there are m users in set U, the time complexity of this step is $O(m^2 * n)$.

Step 3. For each derived indirect friend (at most $m-1$ indirect friends) of target user, we select his/her preferred services (at most n services) and recommend them to the target user. As the time complexity of preferred-service-judgment process is $O(1)$, the time complexity of this step is $O(m * n)$.

With the above analyses, we can conclude that the total time complexity of our proposed Ser_Rec$_{time}$ approach is $O(m^2 * n)$.

5.2. Related Works and Comparison Analyses. With the advent of IoE age, an excessive number of IoE services are emerging on the web, which places a heavy burden on the service selection decision of target users. In this situation, various recommendation techniques, for example, CF-based recommendation [28] and content-based recommendation [29], are put forward to help the target users find their interested IoE services.

A two-phase K-means clustering approach is brought forth in [30] to make service quality prediction and service recommendation. However, this clustering-based approach often requires a dense historical user-service invocation matrix, and hence cannot deal with the service recommendation problem on sparse data very well. In [31], a CF-based recommendation approach is put forward, which realizes service recommendation based on the similar friends of target user. However, when the target user has no similar friends, the recommendation accuracy is decreased significantly. A bidirectional (i.e., user-based CF + item-based CF) service recommendation approach named WSRec is brought forth in [25], for high-quality recommendation results. However, when a target user has no similar friends and similar services, WSRec can only make a rough prediction and recommendation, by considering both the average rating from target user and the average rating of the service that is ready to be predicted. In [26], a MCCP approach is put forward to capture and model the preferences of various users over different services; however, only similar friends of target user are recruited for service quality prediction and recommendation, which drops some valuable user-service relationship information hidden in the historical user-service invocation records.

In order to mine and introduce more user-service relationship information into recommendation, a service recommendation approach SBT-SR is proposed in our previous work [27]. By utilizing "enemy's enemy is a friend" rule in Social Balance Theory, some indirect friends of target user could be found and utilized for further recommendation, which improves the accuracy and recall of recommendation in sparse-data environment. However, SBT-SR has two shortcomings: first, service invocation time is not considered in SBT-SR, which may decrease the recommendation accuracy as service quality is often dynamic and varied with time; second, SBT-SR only employs "enemy's enemy is a friend" rule for service recommendation, while overlooks other valuable rules in Social Balance Theory, for example, "friend's friend is a friend" rule, "friend's enemy is an enemy" rule, and "enemy's friend is an enemy" rule. In view of the above two shortcomings, a novel time-aware service recommendation approach named Ser_Rec$_{time}$ is put forward in this paper, to deal with the recommendation problem in sparse-data

environment. Ser_Rec$_{time}$ considers not only the service invocation time but also the four rules of Social Balance Theory, so that the recommendation accuracy and recall could be ensured. Finally, through a set of experiments deployed on a real web service quality dataset WS-DREAM, we validate the feasibility of Ser_Rec$_{time}$ in terms of recommendation accuracy, recall, and efficiency.

5.3. Further Discussions. In this paper, we put forward a time-aware similarity for service recommendation. Generally, the proposed time-aware similarity could also be applied in other similarity-based application domains, such as content searching [32–36], information detection [37–45], and quality optimization [46–50]. However, there are still several shortcomings in our paper, which are discussed as follows:

(1) The time cost of our proposed Ser_Rec$_{time}$ approach increases fast when the number of users grows. Therefore, the execution efficiency of Ser_Rec$_{time}$ needs to be improved, especially when a huge number of users are present in the historical user-service invocation records.

(2) In this paper, we have investigated the time-aware user similarity. However, besides service invocation time, many other factors, for example, user-service location information, also play an important role in user similarity calculation. In the future, we will improve our proposal by combining the time and location factors together.

6. Conclusions

In this paper, a novel time-aware service recommendation approach, that is, Ser_Rec$_{time}$, is put forward, to handle the service recommendation problems in sparse-data environment where target user has no similar friends and similar services. Instead of looking for similar friends in traditional CF-based service recommendation approaches, in Ser_Rec$_{time}$, we first look for dissimilar enemies of target user based on time-aware user similarity and further determine the indirect friends of target user based on Social Balance Theory. Afterwards, the services preferred by indirect friends of target user are recommended to the target user. Finally, through a set of experiments deployed on a real web service quality dataset WS-DREAM, we validate the feasibility of our proposal in terms of recommendation accuracy, recall, and efficiency.

In the future, we will improve the recommendation effect of our proposal by considering more user-service location information. Moreover, distributed or parallel recommendation approaches will be investigated in the future to improve the recommendation efficiency.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This paper is partially supported by Natural Science Foundation of China (no. 61402258, no. 61602253, no. 61672276, no. 61373027, and no. 61672321) and Open Project of State Key Laboratory for Novel Software Technology (no. KFKT2016B22).

References

[1] Y. Duan, G. Fu, N. Zhou, X. Sun, N. C. Narendra, and B. Hu, "Everything as a service (XaaS) on the cloud: origins, current and future trends," in *Proceedings of the 8th International Conference on Cloud Computing (Cloud '15)*, pp. 621–628, New York, NY, USA, July 2015.

[2] 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.

[3] J. Shen, H. Tan, J. Wang, J. Wang, and S. Lee, "A novel routing protocol providing good transmission reliability in underwater sensor networks," *Journal of Internet Technology*, vol. 16, no. 1, pp. 171–178, 2015.

[4] C. Yuan, X. Sun, and L. V. Rui, "Fingerprint liveness detection based on multi-scale LPQ and PCA," *China Communications*, vol. 13, no. 7, pp. 60–65, 2016.

[5] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Information Sciences*, vol. 295, no. 1, pp. 395–406, 2015.

[6] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340–352, 2016.

[7] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications*, vol. 98, no. 1, pp. 190–200, 2015.

[8] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2594–2608, 2016.

[9] Z. Pan, Y. Zhang, and S. Kwong, "Efficient motion and disparity estimation optimization for low complexity multiview video coding," *IEEE Transactions on Broadcasting*, vol. 61, no. 2, pp. 166–176, 2015.

[10] S. Xie and Y. Wang, "Construction of tree network with limited delivery latency in homogeneous wireless sensor networks," *Wireless Personal Communications*, vol. 78, no. 1, pp. 231–246, 2014.

[11] Z. Zhou, Y. Wang, J. Wu, C. N. Yang, and X. Sun, "Effective and efficient global context verification for image copy detection," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 48–63, 2016.

[12] L. Qi, X. Xu, X. Zhang et al., "Structural Balance Theory-based E-commerce recommendation over big rating data," *IEEE Transactions on Big Data*, 2016.

[13] T. Ma, J. Zhou, M. Tang et al., "Social network and tag sources based augmenting collaborative recommender system," *IEICE Transactions on Information and Systems*, vol. 98, no. 4, pp. 902–910, 2015.

[14] L. Qi, W. Dou, C. Hu, Y. Zhou, and J. Yu, "A context-aware service evaluation approach over big data for cloud applications," *IEEE Transactions on Cloud Computing*.

[15] X. Fan, Y. Hu, R. Zhang, W. Chen, and P. Brézillon, "Modeling temporal effectiveness for context-aware web services recommendation," in *Proceedings of the IEEE International Conference on Web Services (ICWS '15)*, pp. 225–232, New York, NY, USA, June 2015.

[16] S. Wang, L. Huang, C.-H. Hsu, and F. Yang, "Collaboration reputation for trustworthy Web service selection in social networks," *Journal of Computer and System Sciences*, vol. 82, no. 1, pp. 130–143, 2016.

[17] S. Wang, Y. Ma, B. Cheng, F. Yang, and R. Chang, "Multi-dimensional QoS prediction for service recommendations," *IEEE Transaction on Services Computing*, 2016.

[18] Y. Ma, S. Wang, P. C. Hung, C. H. Hsu, Q. Sun, and F. Yang, "A highly accurate prediction algorithm for unknown web service QoS value," *IEEE Transactions on Services Computing*, vol. 9, no. 4, pp. 511–523, 2016.

[19] S. Wang, L. Huang, L. Sun, C.-H. Hsu, and F. Yang, "Efficient and reliable service selection for heterogeneous distributed software systems," *Future Generation Computer Systems*, 2016.

[20] Y. Ni, Y. Fan, W. Tan, K. Huang, and J. Bi, "NCSR: negative-connection-aware service recommendation for large sparse service network," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 579–590, 2016.

[21] Y. Zhong, Y. Fan, K. Huang, W. Tan, and J. Zhang, "Time-aware service recommendation for mashup creation," *IEEE Transactions on Services Computing*, vol. 8, no. 3, pp. 356–368, 2015.

[22] D. Cartwright and F. Harary, "Structural balance: a generalization of Heider's theory," *Psychological Review*, vol. 63, no. 5, pp. 277–293, 1956.

[23] Z. Zheng, Y. Zhang, and M. R. Lyu, "Investigating QoS of real-world web services," *IEEE Transactions on Services Computing*, vol. 7, no. 1, pp. 32–39, 2014.

[24] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the the 10th International Conference on World Wide Web*, pp. 285–295, ACM, Hong Kong, May 2001.

[25] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "QoS-aware web service recommendation by collaborative filtering," *IEEE Transactions on Services Computing*, vol. 4, no. 2, pp. 140–152, 2011.

[26] Y. Rong, X. Wen, and H. Cheng, "A Monte Carlo algorithm for cold start recommendation," in *Proceedings of the 23rd International Conference on World Wide Web (WWW '14)*, pp. 327–336, ACM, Seoul, South Korea, April 2014.

[27] L. Qi, X. Zhang, Y. Wen, and Y. Zhou, "A Social Balance Theory-based service recommendation approach," in *Advances in Services Computing: 9th Asia-Pacific Services Computing Conference, APSCC 2015, Bangkok, Thailand, December 7–9, 2015, Proceedings*, vol. 9464 of *Lecture Notes in Computer Science*, pp. 48–60, Springer, Berlin, Germany, 2015.

[28] F. Zhang, T. Gong, V. E. Lee, G. Zhao, C. Rong, and G. Qu, "Fast algorithms to evaluate collaborative filtering recommender systems," *Knowledge-Based Systems*, vol. 96, pp. 96–103, 2016.

[29] L. Yao, Q. Z. Sheng, A. H. H. Ngu, J. Yu, and A. Segev, "Unified collaborative and content-based web service recommendation," *IEEE Transactions on Services Computing*, vol. 8, no. 3, pp. 453–466, 2015.

[30] C. Wu, W. Qiu, Z. Zheng, X. Wang, and X. Yang, "QoS prediction of web services based on two-phase K-means clustering," in *Proceedings of the IEEE International Conference on Web Services (ICWS '15)*, pp. 161–168, IEEE, New York, NY, USA, July 2015.

[31] S.-Y. Lin, C.-H. Lai, C.-H. Wu, and C.-C. Lo, "A trustworthy QoS-based collaborative filtering approach for web service discovery," *Journal of Systems and Software*, vol. 93, pp. 217–228, 2014.

[32] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Towards efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2706–2716, 2016.

[33] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, "Enabling personalized search over encrypted outsourced data with efficiency improvement," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2546–2559, 2016.

[34] Z. Pan, P. Jin, J. Lei, Y. Zhang, X. Sun, and S. Kwong, "Fast reference frame selection based on content similarity for low complexity HEVC encoder," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 516–524, 2016.

[35] Z. Fu, F. Huang, X. Sun, A. V. Vasilakos, and C. Yang, "Enabling semantic search based on conceptual graphs over encrypted outsourced data," *IEEE Transactions on Services Computing*, 2016.

[36] Z. Fu, X. Sun, S. Ji, and G. Xie, "Towards efficient content-aware search over encrypted outsourced data in cloud," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM '16)*, pp. 1–9, San Francisco, Calif, USA, April 2016.

[37] Z. Xia, X. Wang, X. Sun, and B. Wang, "Steganalysis of least significant bit matching using multi-order differences," *Security and Communication Networks*, vol. 7, no. 8, pp. 1283–1291, 2014.

[38] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507–518, 2015.

[39] Z. Pan, J. Lei, Y. Zhang, X. Sun, and S. Kwong, "Fast motion estimation based on content property for low-complexity H.265/HEVC encoder," *IEEE Transactions on Broadcasting*, vol. 62, no. 3, pp. 675–684, 2016.

[40] Y. Zheng, B. Jeon, D. Xu, Q. M. J. Wu, and H. Zhang, "Image segmentation by generalized hierarchical fuzzy C-means algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 28, no. 2, pp. 961–973, 2015.

[41] B. Chen, H. Shu, G. Coatrieux, G. Chen, X. Sun, and J. L. Coatrieux, "Color image analysis by quaternion-type moments," *Journal of Mathematical Imaging and Vision*, vol. 51, no. 1, pp. 124–144, 2015.

[42] Z. Xia, X. Wang, X. Sun, Q. Liu, and N. Xiong, "Steganalysis of LSB matching using differences between nonadjacent pixels," *Multimedia Tools and Applications*, vol. 75, no. 4, pp. 1947–1962, 2016.

[43] Y. Zhang, X. Sun, and B. Wang, "Efficient algorithm for k-barrier coverage based on integer linear programming," *China Communications*, vol. 13, no. 7, pp. 16–23, 2016.

[44] J. Wang, T. Li, Y. Shi, S. Lian, and J. Ye, "Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics," *Multimedia Tools and Applications*, 2016.

[45] Z. Zhou, C. Yang, B. Chen, X. Sun, Q. Liu, and Q. J. Wu, "Effective and efficient image copy detection with resistance

to arbitrary rotation," *IEICE Transactions on Information and Systems D*, vol. 99, no. 6, pp. 1531–1540, 2016.

[46] L. Qi, W. Dou, and J. Chen, "Weighted principal component analysis-based service selection method for multimedia services in cloud," *Computing*, vol. 98, no. 1-2, pp. 195–214, 2016.

[47] Y. Chen, C. Hao, W. Wu, and E. Wu, "Robust dense reconstruction by range merging based on confidence estimation," *Science China Information Sciences*, vol. 59, no. 9, Article ID 092103, 11 pages, 2016.

[48] Q. Liu, W. Cai, J. Shen, Z. Fu, X. Liu, and N. Linge, "A speculative approach to spatial–temporal efficiency with multi–objective optimization in a heterogeneous cloud environment," *Security and Communication Networks*, vol. 9, no. 17, pp. 4002–4012, 2016.

[49] Y. Kong, M. Zhang, and D. Ye, "A belief propagation-based method for task allocation in open and dynamic cloud environments," *Knowledge-Based Systems*, vol. 115, pp. 123–132, 2017.

[50] B. Gu, V. S. Sheng, and S. Li, "Bi-parameter space partition for cost-sensitive SVM," in *Proceedings of the 24th International Conference on Artificial Intelligence (ICAI '15)*, pp. 3532–3539, Las Vegas, Nev, USA, July 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 4819349, 10 pages
http://dx.doi.org/10.1155/2016/4819349

Research Article

Congestion Control Mechanism for Intermittently Connected Wireless Network

Ruyan Wang, Yang Tang, and Junjie Yan

Chongqing University of Posts and Telecommunications, Chongqing, China

Correspondence should be addressed to Junjie Yan; yd358638469@vip.qq.com

Received 24 September 2016; Accepted 13 November 2016

Academic Editor: Laurence T. Yang

Based on the "storing-carrying-forwarding" transmission manner, the packets are forwarded flexibly in Intermittently Connected Wireless Network (ICWN). However, due to its limited resources, ICWN can easily become congested as a large number of packets entering into it. In such situation, the network performance is seriously deteriorated. To solve this problem, we propose a congestion control mechanism that is based on the network state dynamic perception. Specifically, through estimating the congestion risk when a node receives packets, ICWN can reduce the probability of becoming congested. Moreover, due to ICWN's network dynamics, we determine the congestion risk threshold by jointly taking into account the average packet size, average forwarding risk, and available buffer resources. Further, we also evaluate the service ability of a node in a distributed manner by integrating the recommendation information from other intermediate nodes. Additionally, a node is selected as a relay node according to both the congestion risk and service ability. Simulation results show that the network performance can be greatly optimized by reducing the overhead of packet forwarding.

1. Introduction

Recently, Intermittently Connected Wireless Network (ICWN) has received wide attentions from academia and industry [1]. Due to the sparsity of the node distribution and random movements, a connection between two nodes is dynamic. As a result, the transmitted packets between two nodes can be easily lost, which will lead to the frequent route reestablishment and recovery in ICWN [2]. On the other hand, in practice, the node movement can improve the probability of establishing a connection and thus the network capacity can be improved [3, 4]. Taking advantage of such temporary connections, researchers propose ICWN and design the corresponding architecture [5], in which the nodes except the source and destination nodes can work as the relay. Instead of using the traditional packet forwarding manner (i.e., storing-forwarding), ICWN carries packets in a "storing-carrying-forwarding" way with the help of relay nodes and finally sends packets to reach their destinations.

To realize successful packet transmission and reduce delivery delay, multiple copies of the same packet are injected into ICWN. However, due to limited network resources in ICWN, the buffers of nodes can get saturated quickly. As a result, nodes getting full cannot accommodate more packets, which will result in network congestion. Thus, we have to investigate the network congestion control problem, so as to greatly improve the QoS (Quality of Service) and effectively enhance resource utilization [6].

To address the problem, we propose a dynamic network state perception based on network congestion control mechanism (DNSP-CCM) in this paper. Specifically, we evaluate a node's congestion risk before it receives packets. Particularly, we evaluate the congestion risk in a distributed manner. We also set up the congestion risk threshold that will be adjusted dynamically according to the dynamic network condition. Moreover, we also evaluate a node's service ability, (i.e., a node's message forwarding capability). The service ability can be determined according to the direct encounter probability with other nodes and the indirect encounter probability with the same nodes. We select the node with higher service ability to carry and forward packets. As a result, packets can be transmitted to their destinations in a cost efficient manner so as to effectively alleviate the congestion.

The main contributions of this paper are summarized as follows.

First, we propose a congestion risk evaluation method. The congestion risk level is measured by considering the network status and the buffer size of a node. In particular, the congestion risk threshold is dynamic and changed according to the link condition.

Second, we evaluate a node's service ability. The service ability is evaluated by jointly considering the direct and indirect encounter probability.

Third, we design an adaptive network congestion control strategy based on the congestion risk and service ability. Specifically, we propose an adaptive buffer separation method, (i.e., the forwarding buffer and replacing buffer). According to the transmission status and a link's capacity of a local buffered packets, a node needs to determine the packets needed for forwarding or replacing to enhance the network performance.

The remainder of this paper is organized as follows. Section 2 surveys some research works related to the current congestion control methods. The proposed congestion risk evaluation method is described in Section 3. Section 4 examines an estimation method to measure a node's service capability. Then, an adaptive congestion control mechanism is designed based on the evaluation results of the congestion risk and service capability in Section 5. We show the simulation results in Section 6. Finally, we conclude this paper in Section 7.

2. Related Works

So far, researchers have made great efforts to solve the problem of the network congestion in ICWN. Lo and Lu [7] proposed a mechanism combining with node's neighborhood buffers and node's encounter probability. By obtaining neighborhood nodes' buffer status, a node can dynamically adjust packets quota in order to avoid nodes congestion. This method can alleviate the possibility of changing to congestion status. However, once a node turns into the congestion status, this method only chooses packets to drop by hop counts, which is not reasonable. Besides, only considering the direct encounter probability is unable to well estimate the node's packet forwarding ability.

To make full use of a node's social attributes in ICWN, Daly and Haahr [8] proposed a mechanism which chooses relay nodes based on node relations. However, it ignores the fact that so many nodes are having some relations and thus too many redundant copies of a message occur. Thus, it is easy to cause the network congestion [9]. The node's historical encounter information was utilized to estimate three basic parameters (i.e., the probabilities for head-of-line-blocking, reliability, and deletion in [10]). Nodes make their decisions by these parameters.

On the other hand, researchers considered to use the local congestion status as the network congestion in the area [11]. The forwarding priority is determined according to the degree of data diffusion at the local buffer. Moreover, redundant message copies can be deleted through the active response mechanism. However, the network status cannot be derived from a single node's congestion status. To speed up the data transmission, [12] uses "interest return" and "opportunity consumption" to evaluate the influence on a node's local congestion status and then decide whether to receive messages. This method can alleviate the network congestion and improve network performance to some extent. However, it adopts a fixed congestion threshold, and thus it cannot perceive the really current resource usage.

3. Congestion Risk Evaluation

According to the basic principle of packets forwarding in ICWN, after encountering between two nodes, they need to exchange the packets which are not in the buffer [13]. Obviously, for forwarding multicopy packets, network congestion risk brought by the received packets is directly related to the buffer space. If nodes get more packets, the capacity of continuing to receive packets from other nodes will decrease, resulting in more and more nodes' residual buffer resource decreasing because of the limited buffer resource. If this situation goes on, some areas would generate congestion and then make the whole network congestion. What is more, if nodes have more buffer resources, they can carry more packets to forward, making the probability of dropping packets decrease. So packets can be carried by more nodes which lead to improve delivery ratio. In order to enable nodes to estimate the real-time network congestion level, nodes can achieve the congestion controlling. The risk of network congestion caused by the received packets has to be evaluated.

The packet transmission requires cooperation among multiple relay nodes. Meanwhile, in a given period, the more times a relay node encounters with other nodes, the more chance of diffusion of packets can be gained. Therefore, with the encounter interval T_{interval} between nodes, the larger amount of copies will be injected into the network, which possibly results in higher congestion probability. In addition, the larger Expected Node Meeting Time (ENM), which is defined as a mathematical expectation of nodes encounter interval, the more relay nodes used to carry the packets. Therefore, the forwarding risk level of the forwarding node is defined as follows:

$$R(m) = \frac{\text{ENM}}{T_{\text{interval}}}, \tag{1}$$

where $R(m)$ is the probability of occurring the network congestion when m packets are injected into ICWN. Moreover, the value of T_{interval} can be obtained

$$T_{\text{interval}} = \frac{T_{\text{last}} - T_{\text{first}}}{\beta - 1}, \tag{2}$$

where T_{first} and T_{last} denote the time of first encounter and the last encounter, respectively, between the two nodes, and β denotes the total encounter counts.

As can be seen, in order to get the value of forwarding risk level, nodes need to know encounter time showing the number of times from current time to the next encounter time between two nodes. The law of nodes movement indicates that ENM of nodes in ICWN obeys the exponential distribution with parameter λ and the value of ENM is

$1/\lambda$. It can be described as Generalized Stationary Random Process, whose autocorrelation is only related to time interval τ [14, 15]. Thus, the encounter time of the next future can be estimated by the node's historical information.

Therefore, the expectation of encounter interval between two nodes can be obtained

$$\mathrm{ENM}_X(M \mid t) = \frac{\sum_{s=1}^{k} f_X^s(M)}{\left|\left\{\tau_X^s\left(M_j\right) \geq t\right\}\right|},$$
$$\text{where, } f_X^s(M) = \begin{cases} \tau_X^s(M) - t & \text{if } \tau_X^s(M) \geq t \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $\mathrm{ENM}_X(M \mid t)$ denotes the expectation of encounter interval between node X and node M, t denotes the duration from last encounter to current encounter, $\tau_X^s(M)$ denotes the each interval between nodes encounter, and s denotes the encounter times between nodes.

Using multicopy packet forwarding scheme, a relay node can continuously receive the copies of a packet. As a result, its buffer is getting crowded. If the residual buffer resource is very low, the node is unable to accept newly arriving packets. In this case, we say that this node comes into the congestion status. To evaluate the congestion level, we consider the packet length and available buffer resources. With the longer packet length, the temporary link will be occupied longer. On the other hand, the higher buffer utilization means that the network load is higher, which results in the higher increase in the forwarding risk and congestion risk. Combining both factors, we can obtain the congestion risk $U_{(m,X)}$ before receiving newly arrived packets.

$$U_{(m,X)} = \frac{\mathrm{size}(m)}{\mathrm{AvailableBuffer}(X)} \times R(m). \tag{4}$$

Therefore, combining the average forwarding risk $\overline{R(m)}$, the average packet length $\overline{\mathrm{size}(m)}$, and the occupied buffer $\mathrm{UsedBuffer}(X)$, we can determine the congestion threshold.

$$U_X = \frac{\overline{\mathrm{size}(m)}}{\mathrm{UsedBuffer}(X)} \times \overline{R(m)}, \tag{5}$$

where $\overline{\mathrm{size}(m)}$ and $\overline{R(m)}$ can be obtained as follows:

$$\overline{\mathrm{size}(m)} = \frac{\sum_{l=1}^{\omega} \mathrm{size}(m_l)}{\omega}$$
$$\overline{R(m)} = \frac{\sum_{l=1}^{\omega} R(m_l)}{\omega}, \tag{6}$$

where $\mathrm{size}(m_l)$ denotes the length of packet m_l and ω denotes the number of packets in the buffer.

4. Service Ability Estimation

The node encounter probability and the number of successfully delivered packets are two important parameters used to describe the service capability. Moreover, as ICWN adopts the "storing-carrying-forwarding" method, packets are stored in multiple relay nodes and no direct path is between two nodes. Therefore, we evaluate the packet delivery status by jointly considering the direct and indirect encounter probabilities.

Obviously, the encounter probability can be obtained directly by the encounter times between two nodes. Thus, direct encounter probability of given node pair is shown as follows.

Definition 1. The direct encounter probability between node X and M_j is defined as the ratio of encounter times l *for* node X encountering M_j and the total times n node M_j encountering all other nodes; that is,

$$P_X\left(M_j\right) = \frac{l}{n}. \tag{7}$$

The indirect encounter probability is limited to indirect encounter time interval, average indirect encounter time interval, the total of indirect encounter time interval, and the number of indirect encounters. The indirect encounter time interval means a duration when a node A meets another node, node B, after node A having met another node, node C. For an example, an engineer, considered as node A, goes to company. Before meeting his partner, considered as node B, he may meet with security officer of his company, considered as node C. So node A can help node B to forward packets to node C. The indirect encounter time interval means that the duration node A meets with node C after its meeting with node B. Within a given period of time, the more the number of encounters between nodes, the greater the probability of encounter. Therefore, we use indirect encounter time interval as the estimation parameter of indirect encounter probability. It can make the estimation result more accurate and is conducive to improve the network performance. The definition of indirect encounter probability is shown as follows.

Definition 2. Indirect encounter time interval $T_X(M \mid N)$ denotes the duration from the node X encountering with node M after node X having met with N.

Assume that the meeting time between node A and nodes B and C is recoded as T_B and T_C.

$$T_B = \{T_{B,1}, T_{B,2}, T_{B,3} \cdots T_{B,n}\}, \quad n \longrightarrow \text{connection},$$
$$T_C = \{T_{C,1}, T_{C,2}, T_{C,3} \cdots T_{C,m}\}, \quad m \longrightarrow \text{connection}. \tag{8}$$

Consequently, the total indirect time interval $T_A^S(C \mid B)$ for node A which encounters with node C after meeting with node B can be obtained as

$$T_A^S(C \mid B) = \sum_{k=1}^{m} T_A^k(C \mid B) = \sum_{k=1}^{m} \left(T_{C,r(k)} - T_{B,k}\right), \tag{9}$$

where $r(k) = \min\{i : T_{C,i} \geq T_{B,k}\}$. The average value of $T_A^S(C \mid B)$ can also be obtained as follows:

$$\overline{T}_A(C \mid B) = \frac{T_A^S(C \mid B)}{|r(k)|} = \frac{\sum_{k=1}^{m} \left(T_{C,r(k)} - T_{B,k}\right)}{|r(k)|}, \tag{10}$$

where $|r(k)|$ is the indirect encounter times.

Obviously, the indirect encounter probability is determined by the indirect meeting interval. The lower average value, the higher encounter frequency. Therefore, we use its average value to evaluate the indirect encounter probability in this paper. The definition of indirect encounter probability is shown as follows.

Definition 3. Indirect encounter probability $P_X(M \mid N)$ denotes the probability of node X encountering with node N and consequently encountering with node M.

$$\begin{aligned} P_X(M \mid N) &= \frac{T \times \left|\overline{T}_X(M \mid N)\right|}{T_X^S(M \mid N)} \\ &= \frac{T}{T_X^S(M \mid N) / \left|\overline{T}_X(M \mid N)\right|} \\ &= \frac{T}{\overline{\overline{T}}_X(M \mid N)}, \end{aligned} \tag{11}$$

where T is the given period, $|\overline{T}_X(M \mid N)|$ denotes the indirect encounter time interval, $T_X^S(M \mid N)$ denotes the total indirect encounter time interval, and $\overline{T}_X(M \mid N)$ is the average indirect encounter time interval.

If the encountered node is the destination of a packet, the indirect encounter probability is

$$P_X(M \mid M) = \frac{T}{T_X(M)}, \tag{12}$$

where $T_X(M)$ is the average meeting interval of node X and node M and can be obtained as follows:

$$T_X(M) = \frac{T_{M,\text{last}} - T_{M,\text{first}}}{n-1}, \quad (n = 2, 3, \ldots), \tag{13}$$

where n denotes the encounter times and $T_{M,\text{first}}$ and $T_{M,\text{last}}$ denote the first encounter time and last encounter time between node X and node M, respectively.

In ICWN, each node maintains an encounter information table, within which each item will be updated timely. We show the encounter information table in Table 1, where $M_1, M_2, \ldots, M_j$ are the ID of encountered nodes and $T_{M_1}, T_{M_2}, \ldots, T_{M_j}$ are the every historical encounter time.

Obviously, history information table meets the constraint of complete event group, and each encounter node can be viewed as the corresponding division. Therefore, $P_X(M_j) > 0$, $(i = 1, 2, \ldots, n)$. Taking into account all the divisions of historical encounter event, the delivery probability of packets can be evaluated as

$$\begin{aligned} P_X(D) &= \sum_{j=1}^{K} P_X\left(M_j\right) \times P_X\left(D \mid M_j\right) \\ &= P_X\left(M_1\right) \times P_X\left(D \mid M_1\right) + P_X\left(M_2\right) \\ &\quad \times P_X\left(D \mid M_2\right) + \cdots + P_X\left(M_K\right) \\ &\quad \times P_X\left(D \mid M_K\right), \end{aligned} \tag{14}$$

Table 1: History information.

ID	Connection duration	
M_1	T_{M_1}	$\{T_{M_1,1}, T_{M_1,2}, T_{M_1,3} \cdots T_{M_1,a}\}$
M_2	T_{M_2}	$\{T_{M_2,1}, T_{M_2,2}, T_{M_2,3} \cdots T_{M_2,b}\}$
⋮	⋮	⋮
M_j	T_{M_j}	$\{T_{M_j,1}, T_{M_j,2}, T_{M_j,3} \cdots T_{M_j,y}\}$

where K denotes the number of encountered nodes by node X. Further, the service ability of node x for packet m can be obtained.

$$D_x(m) = \left\lceil P_X(D) \cdot K_X \right\rceil, \tag{15}$$

where K_X is the number of successfully delivered packets. It can be seen that the higher delivery probability can lead to the higher service ability.

5. Adaptive Congestion Control

The congestion threshold should be determined before forwarding packets reasonably. To exploit the resources of a temporary link raised by node movement, the number of packets to be forwarded should be adjusted based on the current network state [16–18]. In this paper, we design an adaptive buffer partition method to well utilize the limited buffer resources. Specifically, the buffer is separate into two parts, that is, forwarding part and replacing part. Moreover, we adjust the separation of the buffer adaptively, highly depending on the transmission status and the link condition [19–22].

Based on the principle of the packet forwarding in ICWN, H_{M_i} means the number of forwarding times which is related to the consumed network resources and has high impact on the congestion status. Thus, for the packets to be forwarded in the buffer, the first i packets, $M_1, M_2, \ldots, M_i$, are injected into the forwarding buffer, while the remaining packets are sent to the replacing buffer. It can be seen that the two parts of the buffer are dynamically adjusted according to the current state. The dynamic threshold is set by the estimated node transmission capacity and the packets state in the buffer.

$$\text{thre} = \frac{\overline{C}_i}{S_{\min}}, \tag{16}$$

where $S_{\min}$ is the minimum length of the buffered packets by node i and $\overline{C}_i$ denotes the estimated transmission capacity for current temporary of node i, which can be obtained as follows.

$$\overline{C}_i = \frac{1}{n}\sum_{r=1}^{n} C_i^r, \tag{17}$$

where C_i^r denotes node i's transmission capacity of rth connection, it can be obtained from the encounter information table, and n is the total encountered times.

On the other hand, to forward the packets to their destinations with less forwarding times, the packets in forwarding

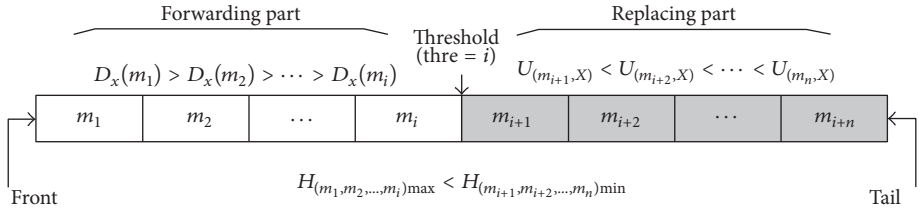


Figure 1: Adaptive buffer partition.

part are sorted in a descending order of the service capability $D_X(m)$. As a result, the higher priorities are assigned for these packets. Additionally, the packets in the replacing part are sorted in ascending order according to their congestion risk $U_{(m,X)}$. The basic principle of adaptive buffer partition is shown in Figure 1.

As can be seen, congestion control process mainly involves two aspects, that is, receiving packets and forwarding packets. For receiving packets, it is essential to determine whether to receive the packet while the residual buffer resources cannot accept newly arrived packet. When deciding to receive packets, the node replaces the packets from the local buffer resource in the region caused by the risk degree of network congestion The buffer operation can solve the problem of the priority of the internal packets forwarding in the buffer [23–26]. According to the basic principle of the ICWN packets forwarding, after encountering two nodes, they need to exchange the congestion state and the packets saved in the buffer [27–30].

The specific operation process of the proposed congestion control mechanism is as follows.

Step 1. When a new packet is generated, the node assigns it with a corresponding identity (Identity, ID) and calculates the forwarding risk $R(m)$.

Step 2. When nodes meet, they exchange their own historical encounter vectors and update their own historical encounter information.

Step 3. When packets are being transmitted between nodes, they are transferred from the head to the tail of the team one by one in the waiting area. Before receiving packet, it is necessary to verify whether the node is packet m's destination. If being verified successfully, then it calculates the local residual buffer to check whether it is sufficient to accommodate the packets. If the buffer is enough, the node receives the packet m directly; otherwise, based on the historical encounter information reserved locally, the node replaces the packet m with the largest congestion risk until it is sufficient to accommodate the packet m. On the other hand, if the local node is not the packet m's destination, then it estimates the packet m's receiving risk $U_{(m,X)}$. If $U_{(m,X)}$ is less than the node's local congestion risk threshold U_X, the packet m's forwarding risk level is reduced by one.

Step 4. If $U_{(m,X)}$ is more than the node's local congestion risk threshold U_X, node's service ability should be calculated. If the local node's service ability for m is stronger than the case of the encounter node, the maintenance of the packet m forwarding risk level cannot be changed and node would receive the packet.

Based on the above description, the time complexity is $O(N^2)$ where N means the total number of buffered packets in a node. Since node should exchange packets and calculate the $R(m)$ and then nodes' service ability, the result may be $O(N^2)$ at the worst situation but can be $O(N)$ at the best situation if all the transmitted packets' destination is the received node and it has enough rooms for these packets.

Data forwarding process pseudocode is shown in Algorithm 1.

6. Numerical Results

In order to evaluate different packet forwarding mechanisms, we use the Opportunistic Networks Environment (ONE) network emulator in this paper to verify our dynamic network state perception based network congestion control mechanism (DNSP-CCM) [31–34]. Further, the performance of DNSP-CCM is compared with several classical mechanisms, First-In First-Drop (FIFD), Last-In First-Drop (LIFD), Drop Least Remaining Life (DLRL), and Drop Most Remaining Life (DMRL).

The performance factors include the delivery ratio, delay, overhead ratio, and load ratio, where the overhead ratio is defined as the proportion between redundant packets forwarding times and the number of successfully delivered P_{overhead} represents the overhead ratio, N_r is the total times of packet forwarding, N_s is the number of successfully transmitted packets [35–38].

$$P_{\text{overhead}} = \frac{(N_t - N_s)}{N_s}. \tag{18}$$

Load ratio is defined as the ratio of the number of messages not successfully delivered and the number of successfully delivered copies, reflecting the network load and transmission overhead.

Besides, two kinds of evaluation methods are utilized, including map-based community model and Infocomm Data model. Map-based community model restricts the movements of nodes to actual streets in an imported map. In our simulation, we use a map of 800 m × 800 m section of Helsinki, Finland. Moreover, the node number is 126, and the transmission range is set to 10 m. Transmission speed

```
(1) node i encounter node j;//
(2) while(connection is up)//
(3) {
(4)    switch summaryvector( i, j);//
(5)    updatenodeserviceability list;//
(6)    updatedatariskrank list;//
(7)    count thre and dividingcacheregion;//
(8)    if( node i is the destination node of new data)//
(9)       while( freebuffersize < newdata.size )
(10)          {remove the data that U is maximum in repacing cache region };//
(11)          { nodei.messagebuffer.receive( new message );//
(12)   else if( newdata.U > node i.buffer.U )//
(13)          { If (nodei.D > nodej.D) //
(14)             {newData.R++;
(15)             nodei.messagebuffer.receive( newmsg );
(16)             }//
(17)             else nodei.buffer.refuse( newdata );//
(18)          }
(19)        else {newdata.R−−; nodei.messagebuffer.receive( newmsg ); }//
(20) }
(21) connectiondown;
```

Algorithm 1: Pseudocode.

for all the nodes is set to 250 KBps. We also assume that the packet length follows exponential distribution within the range of [200, 500] KB. The *Infocom06* data model, which was collected at the 2006 Infocom Conference, contains opportunistic Bluetooth contacts between 98 iMotes, 78 of which are distributed to Infocom06 participants and 20 of which with external antennas (providing longer range) are deployed at several places at the conference venue to act as APs.

6.1. Network Performance under Different Buffer Size. The buffer size has definite influences on load ratio, delivery ratio, and average delay. Figure 2 shows the effects of different buffer sizes on load ratio of five routing mechanisms.

From Figure 2, the load ratio of the five routing mechanisms decreases with the increase of the buffer. The main reason is that, with the increase of the buffer, nodes can carry more messages to improve the probability of successful forwarding while reducing the probability of messages which will be discarded because of buffer overflow. Compared with other mechanisms, the load ratio of DNSP-CCM is the lowest, with 54.2% lower than the of FIFO, 49.9% lower than DLRL, 58.2% lower than LIFD, and 56.5% lower than DMRL. The main reason is that the DNSP-CCM is not only considered the risk of network congestion caused by the messages replacement. In addition, the direct encounter probability and the indirect probability between nodes make the delivery estimation more accurate, choosing the relay node more reasonable and improving the utilization of network resources.

As shown in Figure 3, the delivery ratio of these mechanisms increases with the increase of buffer size. The reason is that, with the increase in buffer capacity, the capacity of node carrying messages increases, and the probability of node congestion level becomes small while the probability of successful delivery is increased. The graph indicates that the delivery ratio of DNSP-CCM is 16.7 higher than that of DLRL, 10.6% higher than FIFO, 22.7% higher than LIFD, and 15.1% higher than DMRL. The results show that DNSP-CCM can select the relay nodes more accurately, reduce the generation of redundant copies, and avoid the part of copies in the forwarding.

As shows in Figure 4, the average delay of the five mechanisms gradually is increased with the increase in buffer size. The main reason is that, with increasing the buffer size, the more messages can be carried by nodes, and the copies can be replaced or discarded in a short time. Therefore, the possibility of the messages in a longer period of time is increased, and then overall average delay is increased. Compared with the other four mechanisms, the performance of DNSP-CCM is the best. This is because that DNSP-CCM can take advantage of direct encounter time interval and indirect encounter time interval between nodes to estimate the nodes direct encounter probability and indirect encounter probability. Thus, the service capability of the nodes is more accurate, and the copies can be successfully delivered in a shorter time.

6.2. Network Performance under Different Packets Generation Interval. The time interval of packets generation can directly reflect the network load. In the case of limited buffer, the shorter the time interval of packets generation, the more likely causing network congestion. The load rate, delivery ratio, and transmission delay of five mechanisms under different buffer sizes are compared from Figures 5–7.

From Figure 5, we can see that the load rate for the five mechanisms is rising with the increasing of the time interval of packets. The main reason is that, with the increase

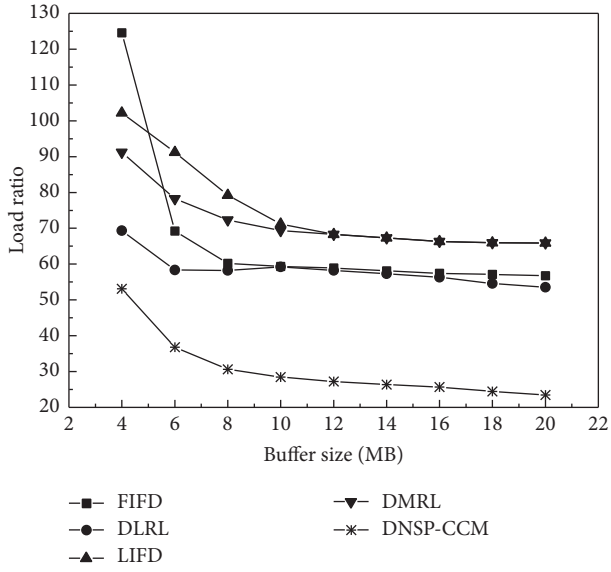


Figure 2: Effects of buffer size on load ratio.

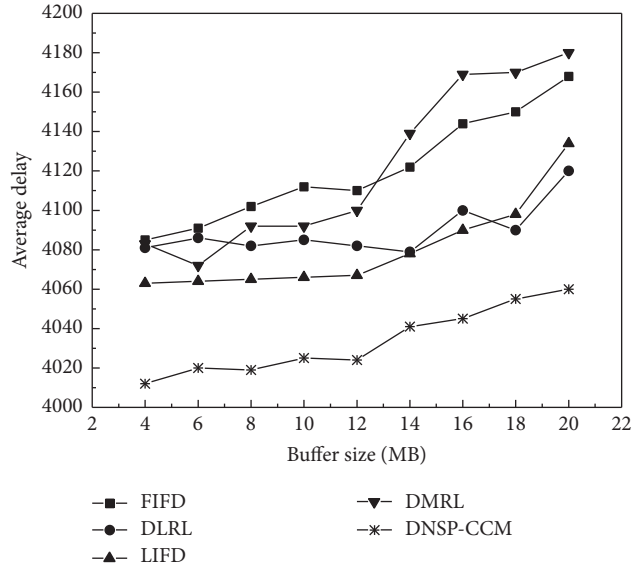


Figure 4: Effects of buffer size on average delay.

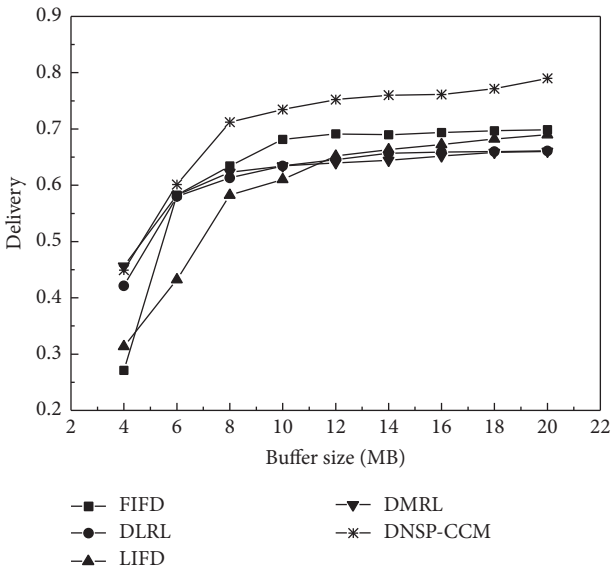


Figure 3: Effects of buffer size on delivery ratio.

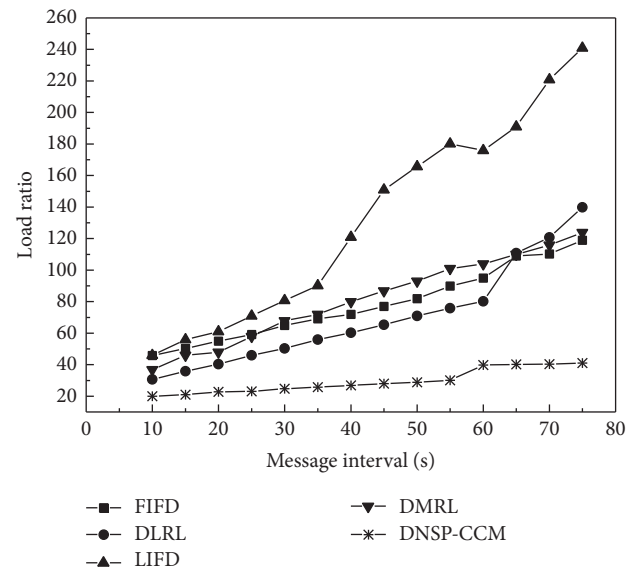


Figure 5: Effects of message interval on load ratio.

in the time interval of packets generation, the total amount of packets generated in the network is reduced. Thus, the number of forwarding packets and the number of successfully delivered packets are declining trend. However, due to the decline in the number of forwarding packets, the trend is smaller than the number of successfully delivered packets, which leads to the increase in the load rate with the increasing time interval of packets generation. Compared with other routing mechanisms, the load rate of DNSP-CCM is the lowest. The load rate of DNSP-CCM was 65.6% lower than that of FIFO, 57.7% lower than DLRL, 69.7% lower than LIFD, and 70.9% lower than DMRL. This is because DNSP-CCM can better use the risk value of the replacement and effectively measure the benefits and disadvantages of forwarding to replace the packets. Thus, making a more accurate decision, avoid the occurrence of network congestion reduces the unnecessary data forwarding and ensure the lower network load ratio.

Figure 6 shows that the delivery ratio of the five mechanisms increases with the increase of the time interval of packets generation. As mentioned above, the main reason is that, with the increase of the time interval of packets, the total amount of packets generated in the network is reduced, and the number of successful deliveries is less than the total amount of packets. Among them, the delivery rate of DNSP-CCM was higher than that of FIFD, 47.8% higher than DLRL, 34.7% higher than, 72.7% higher than LIFD, and 57.2% higher than DMRL. The results show that the

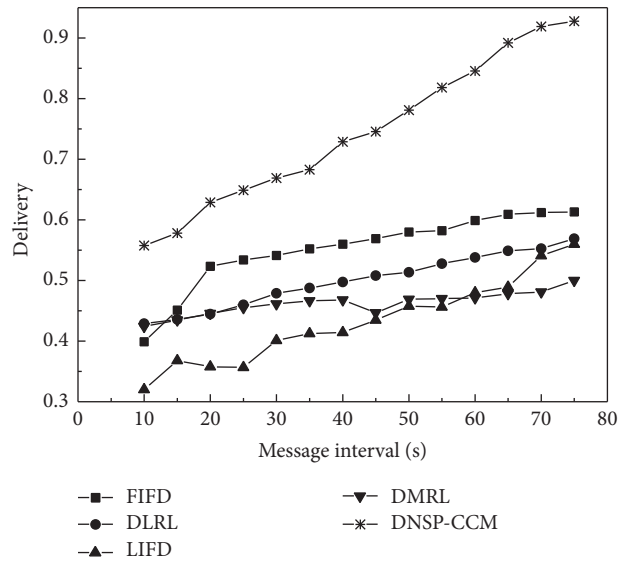


Figure 6: Effects of message interval on delivery.

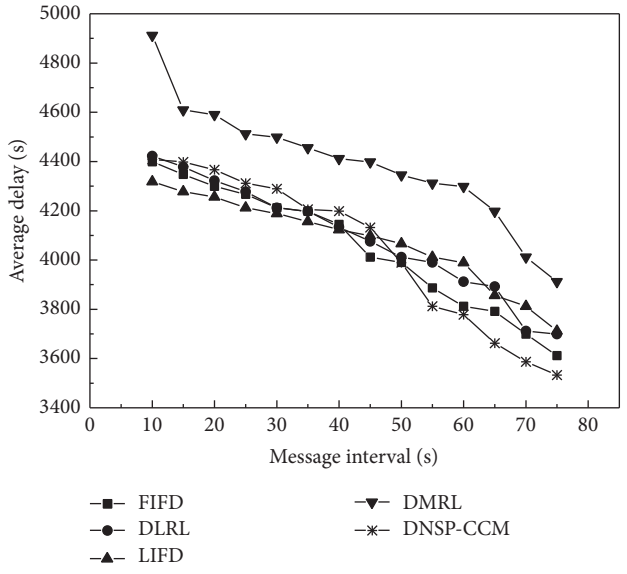


Figure 7: Effects of message interval on delay.

DNSP-CCM can effectively use the packets to replace the risk value of network congestion and reasonably utilize the node's direct encounter probability and the probability of indirect meeting to determine the node service capacity, so as to select the appropriate relay node, avoid the data forwarding process being replaced or discarded, and ensure the successful delivery of data.

Figure 7 demonstrates that the transmission delay of each mechanism decreases with the increase in the time interval of the packets. This is because of the short time in packets generation. Since the amount of packets generated in the network is large, it can easily lead to the overflow of the buffer and reduce the chance of data forwarding in the condition of limited buffer. With the increase in the time interval of the packets, the total amount of packets in the network is reduced, the chance of the packets is forwarded, and the availability of the node buffer resource is increased. Therefore, the network transmission delay is reduced. Among them, the performance of DNSP-CCM is the best, because DNSP-CCM can control network congestion, increase the opportunity of packets delivery, and use the service ability of the nodes to select the intermediate nodes. As a result, the packets can be successfully delivered in a short time.

7. Conclusion

In this paper, we proposed a network congestion control mechanism with the dynamic network state perception to improve the performance of the network. In order to reduce the network overhead and transmission delay, we jointly consider the risk of network congestion caused by node forwarding substitution and the service capability of nodes to accurately evaluate the network state. The proposed DNSP-CCM can improve the packets forwarding capability, control the unnecessary data flooding and network congestion, and then improve the network performance.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (61371097) and Youth Talents Training Project of Chongqing Science & Technology Commission (CSTC2014KJRC-QNRC40001).

References

[1] D. P. Wu, Y. Y. Wang, H. G. Wang, B. R. Yang, C. G. Wang, and R. Y. Wang, "Dynamic coding control in social intermittent connectivity wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7634–7646, 2016.

[2] D. Wu, P. Zhang, H. Wang, C. Wang, and R. Wang, "Node service ability aware packet forwarding mechanism in intermittently connected wireless networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8169–8181, 2016.

[3] M. Peng, Y. Li, Z. Zhao, and C. Wang, "System architecture and key technologies for 5G heterogeneous cloud radio access networks," *IEEE Network*, vol. 29, no. 2, pp. 6–14, 2015.

[4] M. Peng, Y. Li, J. Jiang, J. Li, and C. Wang, "Heterogeneous cloud radio access networks: a new perspective for enhancing spectral and energy efficiencies," *IEEE Wireless Communications*, vol. 21, no. 6, pp. 126–135, 2014.

[5] M. J. Khabbaz, C. M. Assi, and W. F. Fawaz, "Disruption-tolerant networking: a comprehensive survey on recent developments and persisting challenges," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 607–640, 2012.

[6] N. Thompson, S. C. Nelson, M. Bakht, T. Abdelzaher, and R. Kravets, "Retiring replicants: congestion control for intermittently-connected networks," in *Proceedings of the IEEE 29th Conference on Computer Communications (INFOCOM '10)*, March 2010.

[7] S.-C. Lo and C.-L. Lu, "A dynamic congestion control based routing for delay-tolerant networks," in *Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '12)*, pp. 2047–2051, IEEE, Sichuan, China, May 2012.

[8] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant MANETs," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07)*, pp. 32–40, ACM, Quebec, Canada, September 2007.

[9] M. H. R. Khouzani, S. Sarkar, and E. Altman, "Optimal control of epidemic evolution," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 1683–1691, April 2011.

[10] G. Zhang, J. Wang, and Y. Liu, "Congestion management in delay tolerant networks," in *Proceedings of the 4th Annual International Conference on Wireless Internet*, pp. 65–74, ACM, Maui, Hawaii, USA, 2008.

[11] H. Bian and H. Yu, "An efficient control method of multi-copy routing in DTN," in *Proceedings of the 2nd International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC '10)*, pp. 153–156, IEEE, Wuhan, China, April 2010.

[12] Y. Cao and Z. Sun, "Routing in delay/disruption tolerant networks: a taxonomy, survey and challenges," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 2, pp. 654–677, 2013.

[13] D. Wu, J. He, H. Wang, C. Wang, and R. Wang, "A hierarchical packet forwarding mechanism for energy harvesting wireless sensor networks," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 92–98, 2015.

[14] R. Groenevelt, P. Nain, and G. Koole, "Message delay in manet," in *Proceedings of the International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '05)*, pp. 412–413, June 2005.

[15] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Performance analysis of mobility-assisted routing," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '06)*, pp. 49–60, Florence, Italy, May 2006.

[16] Y. Li, Z. H. Zhang, C. G. Wang, W. L. Zhao, and H.-H. Chen, "Blind cooperative communications for multihop ad hoc wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 7, pp. 3110–3122, 2013.

[17] C. Luo, F. R. Yu, H. Ji, and V. C. M. Leung, "Cross-layer design for TCP performance improvement in cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2485–2495, 2010.

[18] Z. Zhang, Y. Yi, and J. Yang, "Energy efficiency based on joint mobile node grouping and data packet fragmentation in short-range communication system," *International Journal of Communication Systems*, vol. 27, no. 4, pp. 534–550, 2014.

[19] Z. Jin, X. Zhao, Y. Luo, and D. Zhao, "Adaptive priority routing with Ack-mechanism for DTN networks," in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '09)*, pp. 1–5, November 2009.

[20] P. Darshana and V. Vandana, "Security enhancement of AODV protocol for mobile Ad hoc network," *International Journal of Application or Innovation in Engineering and Management*, vol. 2, no. 1, pp. 317–321, 2013.

[21] Y. Li, Z. H. Zhang, C. G. Wang, W. Zhao, and H.-H. Chen, "Blind cooperative communications for multihop ad hoc wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 7, pp. 3110–3122, 2013.

[22] Z. Li and H. Y. Shen, "SEDUM: exploiting social networks in utility-based distributed routing for DTNs," *IEEE Transactions on Computers*, vol. 62, no. 1, pp. 83–97, 2013.

[23] C. Wang, B. Zhao, W. Peng, C. Wu, and Z. Gong, "Routing algorithm based on ant colony optimization for DTN congestion control," in *Proceedings of the 15th International Conference on Network-Based Information Systems (NBIS '12)*, pp. 715–720, Melbourne, Australia, September 2012.

[24] N. Thompson, S. C. Nelson, M. Bakht, T. Abdelzaher, and R. Kravets, "Retiring replicants: congestion control for intermittently-connected networks," in *Proceedings of the IEEE INFOCOM 2010*, pp. 1–9, March 2010.

[25] J. Lakkakorpi and P. Ginzboorg, "Ns-3 Module for routing and congestion control studies in mobile opportunistic DTNs," in *Proceedings of the 16th International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS '13)*, pp. 46–50, July 2013.

[26] A. Al-Hinai and H. B. Zhang, "Probabilistic routing based on fine-grained contact characterization in delay tolerant networks," in *Proceedings of the 38th Annual IEEE Conference on Local Computer Networks (LCN '13)*, pp. 581–588, IEEE, Sydney, Australia, October 2013.

[27] R. Das, M. A. T. Prodhan, M. H. Kabir, and G. C. Shoja, "A novel congestion control scheme for delay tolerant networks," in *Proceedings of the International Conference on Selected Topics in Mobile and Wireless Networking (iCOST '11)*, pp. 76–81, October 2011.

[28] B. Soelistijanto and M. P. Howarth, "Transfer reliability and congestion control strategies in opportunistic networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 538–555, 2014.

[29] S. Rashid, Q. Ayub, M. S. M. Zahid, and A. H. Abdullah, "Message drop control buffer management policy for DTN routing protocols," *Wireless Personal Communications*, vol. 72, no. 1, pp. 653–669, 2013.

[30] H. Lu, L. Yin, C. Li, and Y. Wang, "Congestion control in delay tolerant networks with selfish nodes," *Sensor Letters*, vol. 10, no. 8, pp. 1621–1631, 2012.

[31] B. Han, P. Hui, V. S. A. Kumar, M. V. Marathe, J. Shao, and A. Srinivasan, "Mobile data offloading through opportunistic communications and social participation," *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, pp. 821–834, 2012.

[32] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the 2nd International ICST Conference on Simulation Tools and Techniques*, pp. 1–10, March 2009.

[33] S. Rashid, A. Hanan Abdullah, Q. Ayub, and M. Soperi Mohd Zahid, "Dynamic Prediction based Multi Queue (DPMQ) drop policy for probabilistic routing protocols of delay tolerant network," *Journal of Network and Computer Applications*, vol. 36, no. 5, pp. 1395–1402, 2013.

[34] K. Shin, K. Kim, and S. Kim, "Traffic management strategy for delay-tolerant networks," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 1762–1770, 2012.

[35] A. P. Silva, M. R. Hilario, C. M. Hirata, and K. Obraczka, "A percolation-based approach to model DTN congestion control," in *Proceedings of the IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems (MASS '12)*, pp. 100–108, IEEE Computer Society, Dallas, Tex, USA, October 2015.

[36] H. Yan, Q. Zhang, and Y. Sun, "Local information-based congestion control scheme for space delay/disruption tolerant networks," *Wireless Networks*, vol. 21, no. 6, pp. 2087–2099, 2015.

[37] G. Zhao and M. Chen, "A multi-dimensional congestion control mechanism in delay tolerant networks," *Ad-Hoc & Sensor Wireless Networks*, vol. 28, no. 3-4, pp. 319–345, 2015.

[38] A. P. Silva, C. M. Hirata, S. Burleigh, and K. Obraczka, "A survey on congestion control for delay and disruption tolerant networks," *Ad Hoc Networks*, vol. 25, pp. 480–494, 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 5341797, 11 pages
http://dx.doi.org/10.1155/2016/5341797

Research Article

STLIS: A Scalable Two-Level Index Scheme for Big Data in IoT

Yonglin Leng,[1,2] Zhikui Chen,[1] and Yueming Hu[3]

[1]*School of Software Technology, Dalian University of Technology, Dalian 116620, China*
[2]*College of Information Science and Technology, Bohai University, Jinzhou 112100, China*
[3]*College of Natural Resources and Environment, South China Agricultural University, Guangzhou 510642, China*

Correspondence should be addressed to Zhikui Chen; zkchen@dlut.edu.cn

Received 21 August 2016; Accepted 20 October 2016

Academic Editor: Beniamino Di Martino

The rapid development of the Internet of Things causes the dramatic growth of data, which poses an important challenge on the storage and quick retrieval of big data. As an effective representation model, RDF receives the most attention. More and more storage and index schemes have been developed for RDF model. For the large-scale RDF data, most of them suffer from a large number of self-joins, high storage cost, and many intermediate results. In this paper, we propose a scalable two-level index scheme (STLIS) for RDF data. In the first level, we devise a compressed path template tree (CPTT) index based on S-tree to retrieve the candidate sets of full path. In the second level, we create a hierarchical edge index (HEI) and a node-predicate (NP) index to accelerate the match. Extensive experiments are executed on two representative RDF benchmarks and one real RDF dataset in IoT by comparison with three representative index schemes, that is, RDF-3X, Bitmat, and TripleBit. Results demonstrate that our proposed scheme can respond to the complex query in real time and save much storage space compared with RDF-3X and Bitmat.

1. Introduction

The rapid development of the Internet of Thing (IoT) [1] has generated immense sensory data from numerous sensors and devices. In a white paper [2], Cisco predicated that, by 2022, 50 billion things will be connected to the Internet. IDC predicated that 28 billion IoT devices will be installed by 2020 [3]. In facing of such a large number of sensors and devices, IoT data will maintain rapid growth, which poses an important challenge on the storage and quick retrieval of IoT data. RDF (resource description framework) [4], as an effective data representation model, has been applied in many fields including IoT [5]. Therefore, researching the storage and index of large-scale RDF data is very meaningful.

Many works have been engaged in RDF storage and query research. Triple table [6, 7], column storage [8, 9], and property table [10] adopt the alternative storage scheme to accelerate the retrieval of RDF data. But a large number of self-joins and NULL values and scalability constrain the development of these storage systems. RDF-3X [11], Hexastore [12], and SPOVC [13] store the multipermutation of S (subject), P (predicate), and O (object) to match different types of triple patterns. Bitmat [14] and RDFCube [15] use the three dimensions of bit-cube to represent S, P, and O, respectively. Though these methods have a high query efficiency, all of them are at the cost of the storage. Zou et al. [16] proposed a graph-based approach to store and query RDF data (gStore). In order to improve the efficiency of query, gStore creates VS-tree and VS$*$-tree above the RDF graph. TripleBit [17] employs compact technology to store RDF data and introduces two auxiliary index structures to reduce the intermediate results. The index tree and compact technology improve the filter quality and cut down the storage space. However, data filtering and retrieval of gStore and TripleBit are triple-based, which ignore the relation of triple patterns. In order to reduce the number of redundant intermediate results, RP-index (RDF path index) [18] is proposed to efficiently filter data. Wu et al. [19] also used path partitioning to realize the scalable SPARQL query.

To sum up the above discussion, the joins, storage cost, and intermediate results are the main problems in large-scale RDF data storage and index. The tree-based index can effectively filter the redundant data and the path can strengthen the correlation between triples. Therefore, in this

paper, we propose a scalable two-level storage and index scheme based on path partitioning (STLIS), which makes use of the advantages of index tree and the path. The first level of STLIS is used to filter the candidate sets of full path. And the second level is designed to accelerate the match. The main contributions are summarized as follows:

(i) We create a CPTT index based on S-tree to filter the irrelevant RDF data and get the candidate set of full paths. In CPTT index, each leaf node corresponds to a set of full paths, which have the same path tree template.

(ii) During the retrieval process, we design two compressed logical operations, *compressed_AND* and *compressed_OR*, to avoid the decoding operation, thereby improving the efficiency of STLIS.

(iii) We create a HEI index to query the constant predicate path, which can quickly locate the known edge and retrieve it along with the hierarchy of edge backwards or forwards. Through the correlation between triples, HEI dramatically reduce the scale of intermediate results. For the variable predicate path, we design a NP index to convert the variable predicate path into the constant predicate path.

(iv) A set of extensive experiments are executed on a benchmark dataset and real dataset. Results demonstrate that our proposed scheme can respond to the complex query in real time and save much storage space.

The rest of the paper is organized as follows. In Section 2, we review related work about the storage and index of RDF data. Then, in Section 3, we briefly introduce some notations that are closely related to our works, and then we present STLIS index scheme in Section 4. Section 5 provides detailed experimental evaluations compared with the state-of-the-art indexes and Section 6 concludes the paper.

2. Related Work

There are three major kinds of RDF storage and index methods: relational database, triple, and graph.

The management and control technology of relational database is very mature and effective. Many RDF storage systems, for instance, 3-store [6] and Sesame [7], store triples into the relational database. For a SPARQL query, each triple pattern needs to access triple table once, which produces many self-joins and increases query time. In order to improve the retrieval efficiency, many alternative storage schemes are designed to reduce the self-join. Jena2 [10] stores triples into property-class tables that cluster several properties together into a single table. Each property in property-class table is a column and the form is consistent with the relational table. By this way, we can find the triples relating to the same class in a table; therefore, the number of self-joins is reduced dramatically. However, if a SPARQL query involves multiple property-class tables, the merge joins are inevitable. In addition, not all subjects have the same properties in a property-class table, so the property-class table would contain many NULL values. SW-store [8, 9] partitions vertically the triple table into multiple tables according to the predicate. Each table has two columns and all the data in one table have the same predicate. It is easy to query the triple pattern with the given predicate, but this method does not scale well when the predicate is variable.

Triple index scheme is a combination and permutation of S, P, and O. Hexastore [12] creates six different B+tree indexes, SPO, SOP, PSO, POS, OPS, and OSP, and shares common indexes within these 6-way indexes to eliminate storage redundancy. RDF-3X [11] also stores triples into six clustered B+trees above. And, meanwhile, it creates 9 projection indexes: S, P, O, SP, PS, SO, OS, OP, and PO. Instead of storing triples, the projection indexes map search keys to the number of triples. Similar to RDF-3X, instead of 6-way indexes, SPOVC [13] uses 5-way indexes, namely, sIndex (subject), pIndex (predicate), oIndex (object), value, and class, to partition a large triple table into a smaller table. By this way, each triple pattern can be retrieved from a single partition. Bitmat [14] and RDFCube [15] are two kinds of combination indexes based on S, P, and O. Bitmat uses a bit-cube to index the RDF data. The 3 dimensions of bit-cube correspond to S, P, and O, respectively, and each cell corresponds to a unique RDF triple. If the cell is set to 1, the combination of coordinate values of S, P, and O is a given RDF triple; otherwise, the triple does not exist. Similar to Bitmat, RDFCube maps S, P, and O into the 3 dimensions of bit-cube by hash functions. The high efficiency of triple indexes is at the cost of the redundant storage. Though the compression technique is applied to these indexes, the redundancy mode cannot adapt to the data updating and query processing in confronting the large-scale RDF data.

The nature of RDF is a directed graph; therefore, the query can be converted into a subgraph matching problem. Index is an efficient method in the subgraph matching problem, and many indexes based on RDF graph are proposed. GRIN [20] uses a balanced binary tree to index the RDF graph, which groups information around "center" vertices within a given radius. Zou et al. [16] proposed VS-tree and VS$*$-tree index to process both exact and wildcard SPARQL queries by efficient subgraph matching. PIG (parameterizable index graph) [21] is a compact representation of the data graph. Each vertex in PIG corresponds to a group of data graph vertices which have a similar or equal structural "neighborhood." He et al. [22] proposed a bilevel indexing and query processing scheme (BLINKS) for top-k keywords search on graph, which stores summary information at the block level for the search of interblock and more detailed information in each block to accelerate search of intrablock. However, BLINKS only supports the search over node-labeled directed graphs. In order to reduce the number of redundant intermediate results, RP-index (RDF path index) is proposed in [18] to efficiently filter data. TripleBit [17] employs compact technology to store and access RDF data. And, in the query process, TripleBit introduces two index structures to minimize the cost of index selection. To sum up the above discussion, the path, compaction, and index tree are very effective technologies

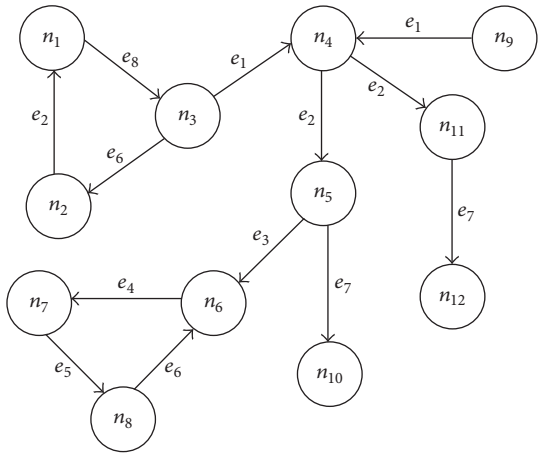


Figure 1: RDF graph.

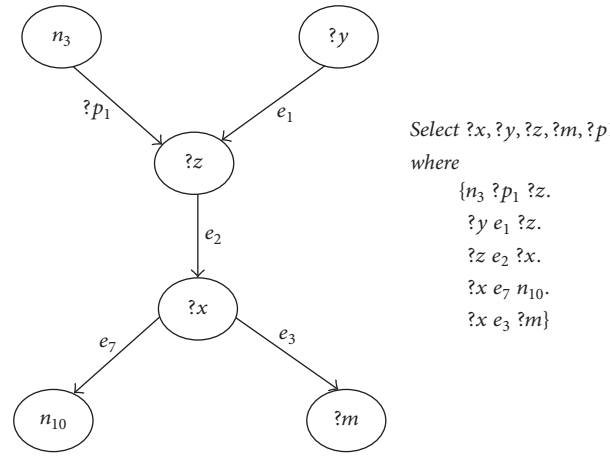


Figure 2: SPARQL query.

in improving the query efficiency and reducing the storage space.

3. Preliminaries

Before going into the details of our scheme, we briefly review some core contexts that are relevant to STLIS.

3.1. RDF Data. In this paper, we use $T = \{t \mid t \in S \times P \times O\}$ to represent a RDF dataset, where S, P, and O are the set of subjects, predicates, and objects, respectively. The corresponding directed label graph of T is denoted as $G = \{V, E, L, f\}$. $V = S \cup O$ is a set of vertices, and $E \subseteq V \times V$ is a set of directed edges from subjects to objects. The label function $f(V \cup E) \rightarrow L$ maps each vertex and edge to a label in L. In the directed label graph, there are two kinds of special vertices: the start vertex and the end vertex. The start vertex refers to the vertex with only outgoing edge or any vertex in a directed cycle in which the directed cycle has no incoming edge, and the vertex with only incoming edge or the start vertex of a directed cycle is the end vertex.

In Figure 1, we use simple notations to substitute URIs and literals. The vertices n_9 and any vertex in the directed cycle (n_1, n_3, n_2, n_1) are the start vertices. The vertices n_{10} and n_{12} and the vertices n_1 and n_6 in the directed cycles (n_1, n_3, n_2, n_1) and (n_6, n_7, n_8, n_6) are the end vertices.

3.2. SPARQL Query. SPARQL is a standard RDF query language [23], which consists of multiple triples and triple patterns. We call the triple including variables the triple pattern. Figure 2 gives an example of SPARQL, where the symbols x, y, z, m, and $p1$ starting with "?" are variables. SPARQL query can also be modeled as graph patterns. In this paper, we use $G_Q = \{V_Q, E_Q, L_Q, f_Q\}$ to describe a SPARQL graph pattern. SPARQL query processing is to find matching triples or subgraphs, in which the information of vertex or edge could substitute for the query variables.

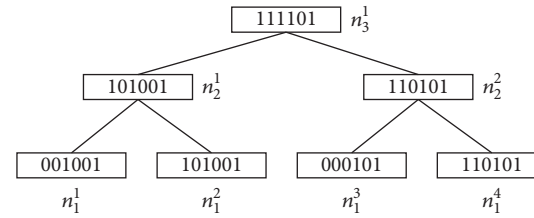
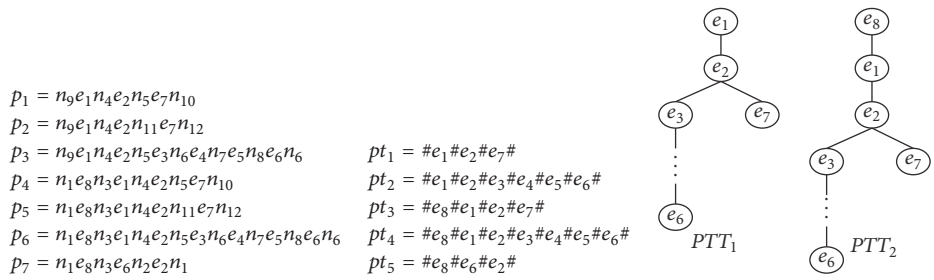


Figure 3: S-tree.

3.3. S-Tree. S-tree is a dynamic balanced multiway signature tree [24], which is similar to B+tree. Each node in S-tree corresponds to one or more signatures. The signature is denoted by a bit-string of fixed length. The bit-string is generated by applying an appropriate hash transformation on the object. The leaf nodes of S-tree are the retrieval objects, and the intermediate nodes are obtained by superimposing the signatures contained in their son nodes. Figure 3 represents a binary S-tree, in which each leaf node corresponds to one signature. n_2^1 is the result of $n_1^1 \mid n_1^2$, where "|" is the bitwise-OR operator. RDF data use Universal Resource Identifiers (URIs) to identify subjects or objects, which are a long string. During the process of retrieval, it is time-consuming to compare strings directly. However, S-tree can use bitwise-AND to realize the comparison quickly. And S-tree can retrieve the object containing variables. Therefore, in this paper, we propose an improved filter index based on S-tree.

4. STLIS

In our index framework, we design a two-level index model. The first level is a filter layer, in which we use CPTT index to filter the RDF data. In order to speed up the efficiency of retrieval and save the storage space, we compress each node in CPTT and give two compressed logical operations, that is, *compressed-AND* and *compressed-OR*, to avoid the decoding operation. The second level is an accurate match layer. We design two indexes, namely, HEI and NP, to assist the retrieval



(a) Full paths (b) Path templates (c) Path tree templates

Figure 4: Full path and path template.

of full path. HEI is fit for the constant predicate path and NP is used in the variable predicate path. In the following sections, we introduce each step in detail. Section 4.1 describes how to get path templates (step 1). Using these path templates, CPTT is created in Section 4.2 (step 2). Then, Section 4.3 discusses how to find the matched path template and the corresponding full path from CPTT (step 3). Finally, we present the index and retrieval of full path in Section 4.4 (step 4).

4.1. Path Template

Definition 1 (full path). Given a RDF graph $G = (V, E)$, if a path $p = v_0e_1v_1e_2v_2 \cdots e_mv_m$ of G satisfies $v_i \in V, e_i(v_{i-1}, v_i) \in E$ and v_0 is a start vertex and v_m is an end vertex. We say that p is a full path. In Figure 4(a), we give all full paths about Figure 1, which start from the start vertices n_1 and n_9.

Theorem 2. *Given a RDF graph $G = (V, E)$, if it is decomposed into a set of full paths $P = \{p_1, p_2, \ldots, p_k\}$ by Definition 1, then each vertex v and edge(u, v) in G can be found in at least one full path.*

Proof. (i) For a vertex $v \in V$ in RDF graph G, there are two kinds of states: start vertex and nonstart vertex. We assume v is a start vertex; then v must belong to a full path. In contrast, if v is a nonstart vertex and there must exist a start vertex s which can reach v, then v belong to a full path starting from s. If there is no source vertex s that can reach v, then v is a start vertex. This contradicts the condition that v is a nonsource vertex. Therefore, for any $v \in V$, it must belong to at least one full path. (ii) For an $edge(u, v) \in E$, if it does not belong to any full paths, then vertex u or v does not exist in any full path. This contradicts (i). So, for any $edge(u, v)$ in G, it can be found in at least one full path. □

Theorem 3. *Given a SPARQL query G_Q that is decomposed into a set of full paths $P_Q = \{p_1^Q, p_2^Q, \ldots, p_n^Q\}$, if G_Q is the subgraph of G, then, $\forall p_i^Q \in P_Q$, there must be at least one full path $p_i \in P$, which satisfies the notion that p_i^Q is a subpath of p_i.*

Proof. Assume there is no full path $p_i \in P$, which satisfies the notion that p_i^Q is a subpath of p_i. (i) If the start vertex v_0 and the end vertex v_m of p_i^Q in G_Q are also the start vertex and the end vertex of G, this will contradict the assumption, because there must be a full path from v_0 to v_m. (ii) If the start vertex v_0 and the end vertex v_m of p_i^Q in G_Q are not the start vertex and the end vertex of G, there must exist a start vertex v_s that can reach v_0 and an end vertex v_e, which is reachable from v_m. Therefore, p_i^Q is a subpath of p_i which starts from v_s and ends up with v_e. □

Definition 4 (path template and homogeneous path). According to Definition 1, a RDF graph G can be decomposed into a set of full paths; that is, $P = \{p_1, p_2, \ldots, p_k\}$. For each path $p_i = v_0e_1v_1e_2v_2 \cdots e_mv_m$, we extract $E(p_i) = \{e_1, e_2, \ldots, e_m\}$ from p_i to make up a schema of path. We say that each schema of path is a path template, and the full paths with the same path template are the homogeneous paths. In Figure 4(b), we give the path templates of Figure 1. The full paths p_1 and p_2 have the same path template pt_1, so we say that p_1 and p_2 are homogeneous paths. The wildcard # represents an instance vertex.

We use a *Depth-First Search* (DFS) algorithm to explore the graph and get the set of full paths and path templates. The algorithm is shown in Algorithm 1. Firstly, the algorithm initializes two stacks: node stack and edge stack (lines (1)-(2)). Secondly, the algorithm finds all full paths and path templates relating to each start node (lines (3)–(18)). For each start node, we push it into node stack (lines (3)-(4)). While the node stack is not empty, we get the top node of stack tn (lines (5)-(6)). If tn does not belong to the set of Sinks, then we use function *neighbor_triples*(tn) to find the unvisited triple associating with tn and push the object and predicate of triple into node stack and edge stack, respectively (lines (13)–(15)). If all of triples associating with tn are visited, we pop up the top node and edge from node stack and edge stack (lines (17)-(18)). Instead, if tn belongs to the set of Sinks, the current node stack and edge stack will be a full path and a path template, respectively. Therefore, we print the node stack and the edge stack and pop up the top node and edge (lines (7)–(12)). The process is repeated, until the node stack is empty.

According to Definition 4, We divide the full paths with the same path template to the same cluster. It is obvious that the number of the full paths will be very huge for a large-scale RDF graph. Therefore, the instance vertex will have many duplicates in different cluster. In order to decrease the duplicates, we propose two merging conditions.

```
Input: Sets Triples, Starts, Sinks
Output: Sets Paths, Path_templates
(1)  Initstack(node);
(2)  Initstack(edge);
(3)  for each s ∈ Starts do
(4)      push(node, s);
(5)      while empty(node) is not NULL do
(6)          tn ← top(node);
(7)          If tn ∈ Sinks then
(8)              Paths ← Paths ∪ print(node);
(9)              Path_templates ← Path_templats ∪ print(edge);
(10)             pop(node);
(11)             pop(edge);
(12)             continue;
(13)         if exist ⟨s, p, o⟩ ∈ neighbor_triples(tn) is not visited then
(14)             push(node, o);
(15)             push(edge, p);
(16)         else
(17)             pop_stack(node);
(18)             pop(edge);
```

Algorithm 1: Full path.

```
Input: matrix pre, set Path_templates, k
Output: set pt_templates
(1)  for each pt ∈ Path_templates do
(2)      initstack(pstack);
(3)      if pt is not merged then
(4)          push(pstack, pt);
(5)          while empty(pstack) is not NULL do
(6)              pt ← pop(pstack);
(7)              for each pt' ∈ path_templates
(8)                  if pre(pt', pt) ≥ k and pt' is not merged then
(9)                      ptt ← merge(pt, pt');
(10)                     push(pstack, pt');
(11)     pt_templates ← pt_templates ∪ ptt;
```

Algorithm 2: Prefix_merge.

Definition 5 (path tree template). Given a set of path templates $PT = \{pt_1, pt_2, \ldots, pt_k\}$, if two or more path templates satisfy the following conditions, one would merge these path templates.

Condition 1. If the path templates share a common prefix, then we merge all these path templates into a path tree template. We use k to represent the length of prefix edge. If the length of prefix edge is equal to or greater than k, then all these path templates are merged. For example, when we set k to 2, the path templates pt_1 and pt_2 would be merged into a path tree template PTT_1, and pt_3 and pt_4 would also be merged into PTT_2. The path tree template is shown in Figure 4(c). It is obvious that the parameter k would affect the number of duplicates of instance vertices. We will discuss this problem in Section 5.4.

Algorithm 2 lists the prefix merge method. The matrix *pre* records the length of common prefix between path templates. We use stack to get each path template set, which satisfies the length of common prefix greater than or equal to k. For each path template pt, if pt is not merged, we find all matching path templates with pt (lines (1)–(7)). And then we merge all these path templates (line (9)). The process is repeated, until all path templates are merged.

Condition 2. Given two path templates pt_a and pt_b, if pt_a is the suffix of pt_b, then we merge pt_a into pt_b. For example, the path template pt_1 is the suffix of pt_3, so pt_1 is merged into pt_3. And PTT_1 and PTT_2 can further be merged into one path tree template PTT_2.

4.2. Compressed Path Tree Template (CPTT) Index. Each path tree template corresponds to a set of full paths. For a SPARQL query, we can retrieve the path template of each query path p_i^Q to get the candidate set of full paths. In this section, we will describe the creation process of CPTT in detail.

Table 1: Statistics of datasets used in experiments.

Dataset	#Vertex	#Edge	#Predicate
LUBM50	1, 706, 230	6, 888, 642	18
LUBM2000	66, 059, 204	276, 345, 040	18
SP^2Bench-100M	548, 826	922, 183	12
UniProt	139,942,781	687,025,165	84

Firstly, we encode each path tree template into bit-string. Secondly, we compress these bit-strings and create a k-way S-tree using these compressed bit-strings. Finally, two logical operations are designed to improve the efficiency of creation and retrieval of index.

Definition 6 (path tree template encoding). Given the predicate set *PRE*, the number of predicates is *plen* and each predicate is assigned to a unique *ID*. The encoding of path tree template is a bit-string of length *plen* and each bit corresponds to a predicate. Assume that PE_i is the set of predicates of path tree template pt_i; if $\exists pre \in PE_i$, then one sets the *eId(pre)*th bit of bit-string to "1," where *eId*(*pre*) is used to get the *ID* of *pre*. In Figure 1, the number of predicates is 8, so the bit-string of pt_1 is "11000010."

Table 1 gives the statistics of predicate about the benchmark datasets and real dataset. The number of predicates is much smaller than the number of vertices and edges. Hence, the encoding scheme is feasible. Also, the bit-string exhibits many consecutive 0 or 1 values, so a compressed method can be used to further decrease the storage space of bit-string. In this paper, we use Run Length Encoding (RLE) to achieve the gap compressed representation of the bit-string. A bit-string of pt_1 will be represented as "[1] 2 4 1 1" in which the first bit of compressed bit-string represents the first bit value and the other bits record the alternating run lengths of 0 and 1.

Given a set of compressed bit-strings, we build an index over all these compressed bit-strings, which is called CPTT index (compressed path tree template index). The build and search process of CPTT is similar to S-tree. Bitwise-AND and bitwise-OR are two basic operations of S-tree. In order to avoid the transformation back and forth between compressed encoding and bit-string, in this paper, we propose the *compressed-AND* and *compressed-OR* operations. Algorithm 3 represents the *compressed-AND* operation. In *compressed-AND*, we adopt the replace-alignment method to compute the result. From the first corresponding compressed bit, we choose the greater compressed bit and replace it with the smaller compressed bit and the difference between the two compressed bits. And then we compute the second bit, and so forth, until all compressed bits have been checked. For example, two compressed strings A = [1] 2 4 2 and B = [0] 3 1 4, in which A-2 is less than B-3, so we replace B-3 with B-2 1. And then B = [0] 2 1 1 4. For the second bit, we compare A-4 and B-1 with the same method, and so on. Due to the fact that OR operation can be denoted as A OR B = NOT(NOT(A) AND NOT(B)) and NOT only needs to change the first value of the compressed bit-string,

```
(1)  sign_a ← Get_sign(A); sign_b ← Get_sign(B);
(2)  while i < A.size() and j < B.size() do
(3)      if Get_str(A, i)-Get_str(B, i) > 0 then
(4)          Replace(A, i, Get_str(B, i));
(5)          Insert(A, i + 1, Get_str(A, i)-Get_str(B, i));
(6)          Change_sign(sign_b);
(7)      else if Get_str(A, i)-Get_str(B, i) < 0 then
(8)          Replace(B, j, Get_str(A, i));
(9)          Insert(B, j + 1, Get_str(B, i)-Get_str(A, i));
(10)         Change_sign(sign_a);
(11)     else
(12)         Change_sign(sign_a);
(13)         Change_sign(sign_b);
(14)     i = i + 1; j = j + 1;
(15) return Merge(A, B);
```

Algorithm 3: *Compressed_AND*(A, B).

therefore, we can use the *compressed-AND* to achieve the result of *compressed-OR*.

We use a bottom-top process to create the CPTT index. Each leaf node of CPTT is a path tree template, and each path tree template corresponds to a set of full paths. Each intermediate node is got by superimposing the compressed bit-strings contained in its son nodes. Figure 5 gives an example of CPTT (in order to explain the retrieval process, Figure 5 is created based on the path template of Figure 4(b)).

4.3. Retrieval of CPTT

Definition 7 (match rule). Given a compressed bit-string of path tree template pt^* and a query path compressed bit-string qt^*, pt^* is a match of qt^*, if and only if *compressed-AND*$(pt^*, qt^*) = qt^*$.

Note that, in the path template, if there are variable edges, we only consider the constant edges. For example, in Figure 2, the path template of query path $n_3?p_1?ze_2?xe_7n_{10}$ is $\#?p_1\#e_2\#e_7\#$. $?p_1$ is a variable; therefore, we discard $?p_1$ and only encode the edge of e_2 and e_7 into the bit-string, that is, 01000010. To find the match path tree template over a given CPTT index, we adopt a top-down search strategy. Due to the fact that the root and intermediate node of CPTT are the summary of their corresponding sons, therefore, if qt^* does not match the root or intermediate node, there must not exist the match path template of qt^*. Consequently, we can filter the path tree template and the corresponding paths. For example, in Figure 5, we want to scan the path template "$\#e_1\#e_2\#e_7\#$" whose compressed encoding is [1] 2 4 1 1. The red dash line represents the retrieval path. We can see that the intermediate node n_3^1 does not satisfy Definition 7. Therefore, all descendant nodes do not need to be retrieved. The final candidate path templates are pt_1 and pt_3. If the path template is "$\#?p_1\#e_1\#e_2\#$" then the candidate results are pt_2, pt_4, pt_1, and pt_3. From these, we can see that the exact path template information and the independent path edge set would get a better filter result. Therefore, the Hamming

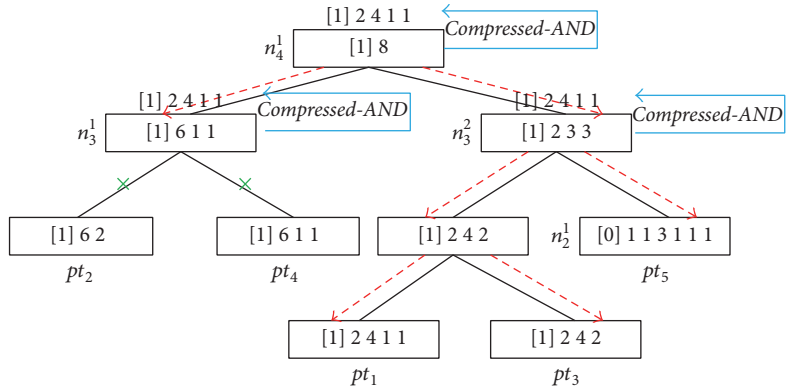


Figure 5: Example of CPTT.

Input: the query path template qt^*, the index of CPTT
Output: a set of match path templates PT

(1) $PT \leftarrow \varnothing$;
(2) $Initstack(ST)$;
(3) $pt^* \leftarrow root(CPTT)$;
(4) $push(ST, pt^*)$;
(5) while ST is not empty
(6) $\quad pt^* \leftarrow pop(ST)$
(7) $\quad$ if $compressed\_AND(pt^*, qt^*) = qt^*$
(8) $\quad\quad$ for each child pt_c^* of pt^* do
(9) $\quad\quad\quad$ if pt_c^* is not a leaf node
(10) $\quad\quad\quad\quad push(ST, pt_c^*)$;
(11) $\quad\quad\quad$ else
(12) $\quad\quad\quad\quad$ if $compressed\_AND(pt_c^*, qt^*) = qt^*$
(13) $\quad\quad\quad\quad\quad PT = PT \cup pt_c^*$;
(14) return PT

Algorithm 4: Retrieval on the CPTT.

distance is used to calculate the similarity between two bit-strings. If the Hamming distance between two bit-strings is the smallest, the two bit-strings will be siblings. The retrieval operation is described briefly in Algorithm 4.

Algorithm 4 uses stack to achieve the retrieval. Firstly, we initialize the stack ST and push the root node into ST (lines (2)–(4)). Next, while ST is not empty, we get the top element pt^* (lines (5)-(6)). We compute the *compressed_AND* between pt^* and qt^* (line (7)). If the result is qt^*, then we check all the child nodes pt_c^* (line (8)). If pt_c^* is not a leaf node, we push pt_c^* into ST, or else we further compute the *compressed_AND* between pt_c^* and qt^*. If the result is qt^*, we merge pt_c^* into PT (lines (9)–(13)). The process is repeated, until the stack ST is empty.

4.4. Index and Retrieval of Full Path. In order to obtain the final query results, we further define a hierarchical edge index (HEI) and a node-predicate (NP) index to assist the retrieval of full path. In this paper, we divide the query path into two categories: the constant predicate path and the variable predicate path. The constant predicate path refers to the notion that one or more predicates in the query path are known. Instead, the variable predicate path refers to the notion that all predicates in query path are unknown. For the constant predicate path, we use CPTT to get the candidate full path sets which contain the known predicates of path edge. The constant predicate path is the most common. We observe a mount query in real RDF datasets, in which most of the edges are known. For example, the edges of query for UniProt datasets in [17] are all constants. So, CPTT index is very effective for pruning the full path.

In the candidate full path sets, we further execute the node match to get the final results. Next, we will introduce the hierarchical edge index (HEI), including the index structure, storage scheme of triples, and the retrieval procedure. HEI contains all edges of path tree template and each edge corresponds to a set of triples and a hierarchical *ID*. As RDF data use Universal Resource Identifiers (URIs) to identify subjects or objects, then, in order to save the storage space and improve the retrieval efficiency, we replace all URIs with *IDs* by mapping dictionaries (see, e.g., [17]). In this paper, we assign *IDs* to subjects and objects with the same *ID* such that subjects and objects having identical values will be treated as the same entity.

Because the query path may be a subset of full paths and some predicates of path edge are unknown, therefore, the retrieval of query path may not start from the root of path tree template. In order to locate the known edge quickly, HEI sequentially indexes each edge of path tree template and assigns a hierarchical *ID*. Meanwhile, each index edge also relates to a set of triples. To save storage space, each triple is encoded with variable-size bits and compressed by RLE. Given a triple, we select the maximum number of bits that encode the *ID* of subject or object as the length of encoding of subject and object. Considering the example of triple $\langle 21, p1, 267\rangle$, the encoding of the subject needs 1 byte and that of the object needs 2 bytes. Therefore, we select 2 bytes as the length of encoding of $\langle 21, p1, 267\rangle$, for example, 0000000000010101, 0000000100001011. Due to the large number of consecutive 0 or 1 values, therefore, we adopt RLE to achieve the gap compressed representation of the bit-string. Figure 6 gives an example about the index and storage of HEI.

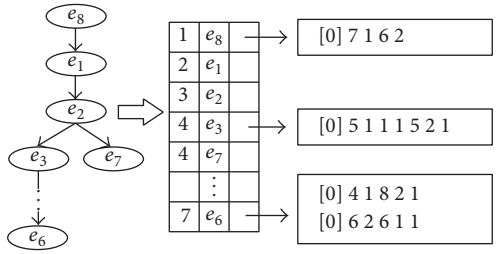


Figure 6: Example of HEI.

For a constant predicate path, the query step is as follows: (1) selecting a constant predicate, locate the predicate edge in HEI; (2) encoding the triple pattern of constant predicate with the same encoding method mentioned above, if a variable exists in the triple pattern, the encoding of corresponding variable is set to "0"; the encoding of the triple pattern also needs to be compressed; (3) use *compressed-AND* to compare the triple pattern with the set of corresponding edge triples; (4) if the match result exists, the retrieval would continue along the path backwards and forwards with the assist of hierarchical *ID*.

Through CPTT, the paths that do not meet the match conditions are filtered. HEI uses the constant predicate to retrieve the unknown information of triple, which provides more related information for the adjacent triples. By this way, the joins among triples decrease dramatically. However, the above method only suits the constant predicate path; in face of the variable predicate path, CPTT is invalid. However, the variable predicate path is extremely rare in the SPARQL of a real dataset. But, in order to ensure the whole efficiency of retrieval, we define an auxiliary index, that is, NP index, to solve this problem.

NP index is a bit-matrix, in which the row corresponds to the set of nodes and the column represents the set of predicates. If a node relates to a predicate, then we set the corresponding bit to be "1." For all the triples of a query path like $\langle s, ?p, ?o\rangle$, $\langle ?s, ?p, o\rangle$, and $\langle s, ?p, o\rangle$, we locate the constant node s or o in bit-matrix to get the related predicate p. Next, all the predicates p in the query path are encoded into a compressed bit-string according to Definition 6. Using this compressed bit-string, we can query other unknown information by CPTT and HEI.

5. Experiments

In this section, we compare STLIS scheme with some popular index schemes, including RDF-3X (v0.3.8), Bitmat, and TripleBit, on both synthetic and real datasets.

5.1. Datasets and Setting. To evaluate the performance of STLIS, two representative RDF benchmarks, LUBM (the Lehigh University Benchmark) [25] and SP^{2}Bench [26], are used in our experiments. The LUBM features a university domain, and the SP^{2}Bench dataset features a DBLP domain. A real dataset, UniProt, is also utilized in our experiments, which is a protein dataset [27]. Table 1 shows the detailed information about each dataset. For query information about LUBM, refer to Appendix A.2 in [28]; for UniProt, see [17]; and for SP^{2}Bench, see [26].

Table 2: Query response time on LUBM50 (seconds).

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
RDF-3X	3.09	0.37	2.34	**0.45**	0.47	1.38	3.28
Bitmat	3.51	0.51	2.37	1.03	0.59	2.69	3.95
TripleBit	2.43	**0.25**	1.07	0.86	**0.43**	0.79	0.66
STLIS	**0.38**	0.42	**0.29**	0.78	0.53	**0.19**	**0.57**

Table 3: Query response time on LUBM2000 (seconds).

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
RDF-3X	280.65	4.179	153.91	5.036	3.67	10.344	143.35
Bitmat	15.48	0.96	9.56	1.54	**1.24**	5.98	9.78
TripleBit	23.029	3.062	6.38	6.056	5.011	12.105	35.799
STLIS	**6.67**	**0.83**	**4.54**	**1.03**	1.44	**2.37**	**3.38**

STLIS index is implemented using C++ and compiled using G++, with-O2 option for optimization. The experiments are performed on a PC with Intel Xeon at 2.00 GHz × 24, with 20 GB memory, running 64-bit Linux. To account for caching, each of the queries is executed three times consecutively. We take the average result to avoid artifacts caused by OS activity.

5.2. Performance. During the construction process of STLIS, there are 81 species path templates in LUBM dataset. We set the number of prefix edges k to 3 and get 26 kinds of merged path tree templates. We run the same queries of LUBM as [28] did, and the query response times are listed in Tables 2 and 3. We can see that the query response time of STLIS outperforms the other three index structures, especially the complex queries. The primary reasons lie in the filter condition firstly. CPTT is a combination index of path edges. The relation among edges enhances the filter ability. Therefore, a large number of unrelated triples are filtered, which dramatically reduces the query scale.

Secondly, the intermediate result is another important cause. Regarding intermediate results, we refer to both the number of triples that match the query patterns and the data loaded into the main memory during query evaluation [17]. Due to the compressed encoding and compressed operations, our scheme improves the memory utilization and reduces the I/O cost. Furthermore, in the further retrieval of full path, most of the joins among triples happen in the interior of path retrieval, which leads to less intermediate result for final joins.

As shown in Tables 2 and 3, STLIS is very effective on large-scale dataset. The query response time of RDF-3X increases 7 to 90 times from LUBM50 to LUBM2000, especially the complex queries (Q1, Q3). But the maximal change of STLIS is only 17.55 times. The reasons lie in the fact that RDF-3X needs to load more permutation indexes into the memory for scan. Moreover, the load data need to be decompressed. Therefore, the I/O is heavier in RDF-3X and much time is consumed on decompressed data. The

Table 4: Query response time on UniProt (seconds).

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
RDF-3X	75.11	1.95	337.28	24.36	15.64	169.76	45.34	28.36
TripleBit	14.27	**1.05**	112.48	6.68	8.29	6.86	13.65	17.34
STLIS	**9.68**	1.34	**13.56**	**4.38**	**3.27**	**4.28**	**5.67**	**1.24**

Table 5: Query response time on SP^{2}Bench-100M (seconds).

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
RDF-3X	0.053	0.46	0.072	0.32	0.095	0.27	0.69	0.77
Bitmat	0.028	0.39	0.064	0.26	0.14	0.19	0.53	0.59
TripleBit	0.014	0.27	0.043	0.27	0.078	0.16	0.42	0.44
STLIS	**0.008**	**0.03**	**0.007**	**0.07**	**0.016**	**0.04**	**0.09**	**0.03**

Table 6: Storage space (GB).

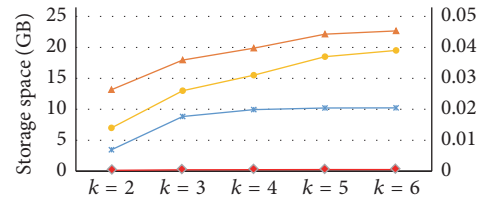
	RDF-3X	Bitmat	TripleBit	STLIS
LUBM50	0.35	0.32	0.28	0.22
LUBM2000	13.95	13.64	8.74	8.83
UniProt	33.89	55.83	15.19	17.95
SP^{2}Bench-100M	0.038	0.044	0.034	0.026

queries of Bitmat and TripleBit are all based on triples, which produce more intermediate results than STLIS, so the query performance is also lower than STLIS.

For the UniProt dataset, the triple scale is up to 0.7 billion and the number of prefix edges k is equal to 3. Bitmat cannot get the query results using the current operation environment. Therefore, Table 4 only lists the comparison among RDF-3X, TripleBit, and STLIS methods. We can see that STLIS outperforms RDF-3X and TripleBit, especially the complex query. The reason lies in the fact that STLIS decomposes the query into paths. Each path edge set is seen as a filter condition. Compared with the triple-based filter, STLIS gets a higher quality candidate set. In the candidate set, the joins between triples only occur in the interior of a query path, which effectively reduce on the scale of intermediate results. Therefore, our query performance is superior to other methods.

We also execute the queries on SP^{2}Bench dataset. The representative queries in [26] are selected to evaluate the query response time. The same as LUBM, we also set k to 3. The experiment results are listed in Table 5. Due to the fact that SP^{2}Bench dataset is very small, the intermediate results of most of the queries can reside in memory directly; therefore, the advantage of STLIS is not obvious.

5.3. Storage Space. We compare the storage space of STLIS with RDF-3X, Bitmat, and TripleBit. The total space cost refers to the size of the whole database and indexes. Besides Bitmat, the other three storage spaces all include the dictionary facility. The detailed comparison is listed in Table 6. We can see that STLIS outperforms RDF-3X in all datasets.



	k = 2	k = 3	k = 4	k = 5	k = 6
LUBM2000	3.45	8.83	9.94	10.21	10.23
UniProt	13.17	17.95	19.87	22.13	22.67
LUBM50	0.15	0.22	0.25	0.26	0.27
SP^{2}Bench-100M	0.014	0.026	0.031	0.037	0.039

Figure 7: The comparison of storage spaces in parameter k.

The reason is that RDF-3X creates six clustered indexes and 9 aggregate indexes. The high efficiency of RDF-3X is at the cost of space. In addition, with the data scale growth, the storage of RDF-3X and Bitmat grows faster than STLIS. Thus, we conclude that STLIS is more suitable for large-scale dataset. In this paper, we adopt the same dictionary mapping algorithm as TripleBit. As the predicate numbers of LUBM and SP^{2}Bench are very small, in consequence, the merged quality of path template is very high, leading to a smaller duplicate. So, STLIS is comparable with TripleBit. However, for UniProt, the storage space of STLIS is higher than TripleBit.

5.4. Parameter Analysis. In this section, we evaluate the effect of parameter k. In the process of merge path template, we use k to represent the number of common prefix edges. If the number of prefix edges is equal to or greater than k, then all path templates including common prefix edges are merged.

We vary k from 2 to 6 to test the change of storage space. Figure 7 shows that the storage spaces of all datasets increase when k becomes larger. This is because the larger the value of k is, the smaller the common prefix path templates are, therefore leading to the increase of duplicates. However, with the increase of k, the fluctuation of storage space becomes smaller. As shown in Figure 7, while the value of k is equal to 4 or 5, the storage space tends to stability for LUBM and SP^{2}Bench. But, for UniProt, the range is between 5 and 6. The reason is that the number of predicates of UniProt is larger than other datasets.

Figure 8 is the comparison of query response times. The experiment result demonstrates that k has an optimum value which usually is not the biggest or the smallest in query response time. For example, in LUMB and SP^{2}Bench dataset, k is equal to 3 and the value of k is 4 in UniProt. The primary reasons lie in the fact that if k is set to a small value, a large number of path templates would be merged into a path tree template. The candidate set of full paths becomes very large, which would increase the query time. Instead, we select a large value, though the candidate set is smaller than the above selection method, but the increase of duplicates also consumes the query time.

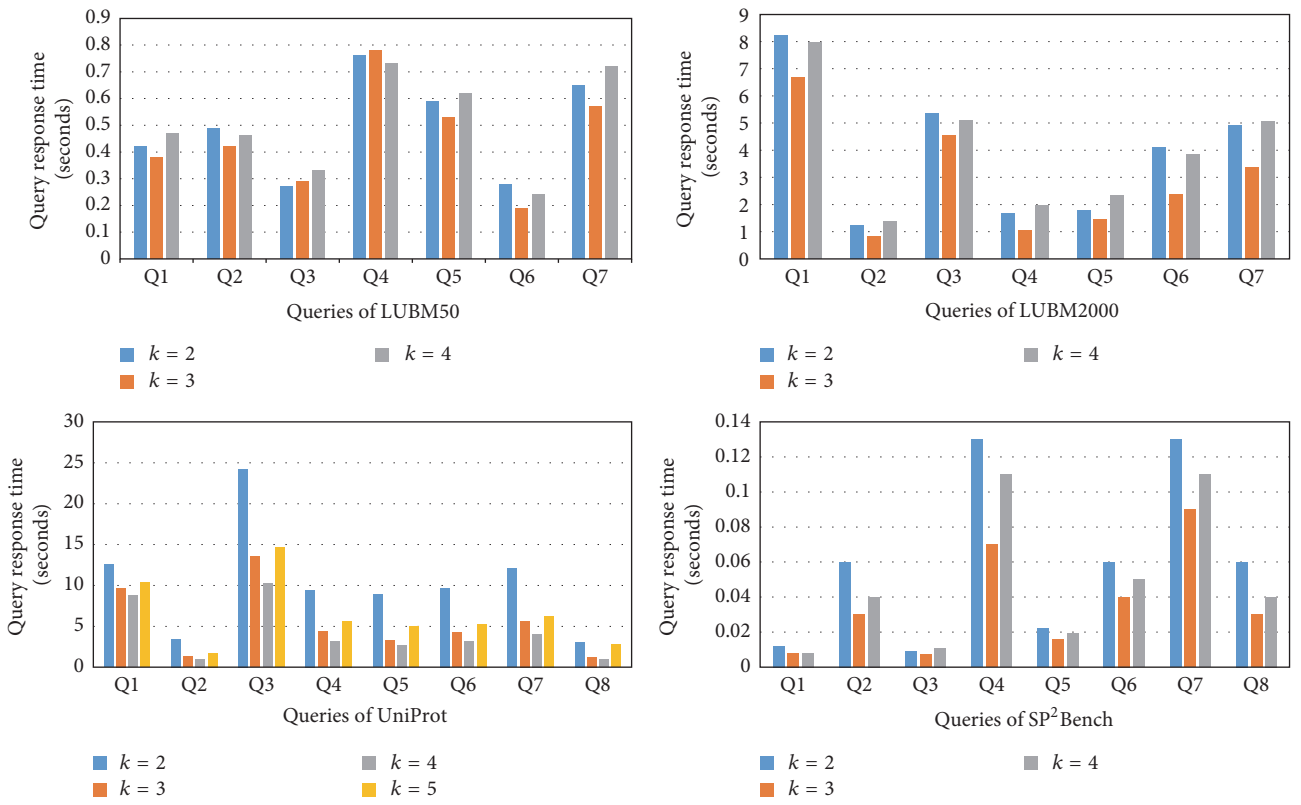


Figure 8: The change of query response time in parameter k.

6. Conclusion

In this paper, we presented a scalable two-level index scheme (STLIS) for big data in IoT. The first level is a filter layer, in which we used CPTT index to get the candidate set of full paths. In order to speed up retrieval and save storage space, each node of CPTT was a compressed bit-string. And, to avoid the decoding operation, we gave two compressed logical operations, that is, *compressed-AND* and *compressed-OR*. The second level was an accurate match layer. In this layer, we designed two auxiliary indexes, that is, HEI and NP, to assist the retrieval of full path. HEI fitted for the retrieval of constant predicate path and NP was used in variable predicate path. Experimental results demonstrated that our proposed scheme can respond to the complex query in real time, and it was effective to reduce the storage space by encoding the *ID*s of subject and object with variable-size bits and compressed technologies.

Furthermore, distributed index and query is the most effective scheme to deal with the big increasing number of IoT data. Our index scheme is very scalable for the distributed environment. CPTT index can be partitioned into several subtrees; each subtree corresponds to a set of leaf nodes and each leaf node is related to a set of full paths. Those subtrees can be distributed to different computing nodes of a compute cluster. Thus, queries can be executed in parallel using a distributed framework.

Competing Interests

The authors declare that there are competing interests regarding the publication of this paper.

Acknowledgments

This work is partially supported by the State Key Program of National Natural Science of China under Grant U1301253, the Science and Technology Planning Key Project of Guangdong Province under Grant 2015B010110006, and the National Natural Science Foundation of China under Grant 61672123.

References

[1] D. Singh, G. Tripathi, and A. J. Jara, "A survey of Internet-of-Things: future vision, architecture, challenges and services," in *Proceedings of the IEEE World Forum on Internet of Things (WF-IoT '14)*, pp. 287–292, March 2014.

[2] J. Bradley, J. Barbier, and D. Handler, "Embracing the Internet of everything to capture your share of $14.4 trillion," White Paper, Cisco, 2013.

[3] IDC Market in a Minute: Internet of Things, http://www.idc com/downloads/idc_market_in_a_minute_iot_infographic.pdf.

[4] RDF, http://www.w3.org/TR/rdf-concepts/.

[5] X. Su, H. Zhang, J. Riekki et al., "Connecting IoT sensors to knowledge-based systems by transforming SenML to RDF," *Procedia Computer Science* , vol. 32, pp. 215–222, 2014.

[6] S. Harris and N. Gibbins, "3store: efficient Bulk RDF Storage," in *Proceedings of the 1st International Workshop on Practical and Scalable Semantic Systems*, pp. 1–15, 2003.

[7] J. Broekstra, A. Kampman, and F. van Harmelen, "A generic architecture for storing and querying rdf and rdf schema," in *The Semantic Web—ISWC 2002: 1st International Semantic Web Conference Sardinia, Italy, June 9–12, 2002 Proceedings*, vol. 2342 of *Lecture Notes in Computer Science*, pp. 54–68, Springer, Berlin, Germany, 2002.

[8] D. J. Abadi, A. Marcus, S. R. Madden et al., "Scalable semantic web data management using vertical partitioning," in *Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB '07)*, pp. 411–422, VLDB Endowment, September 2007.

[9] D. J. Abadi, A. Marcus, S. R. Madden, and K. Hollenbach, "SW-Store: a vertically partitioned DBMS for semantic web data management," *The VLDB Journal*, vol. 18, no. 2, pp. 385–406, 2009.

[10] J. J. Carroll, I. Dickinson, C. Dollin et al., "Jena: implementing the semantic web recommendations," in *Proceedings of the 13th ACM International World Wide Web Conference on Alternate Track Papers & Posters (WWW '04)*, pp. 74–83, New York, NY, USA, May 2004.

[11] T. Neumann and G. Weikum, "RDF-3X: a RISC-style engine for RDF," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 647–659, 2008.

[12] C. Weiss, P. Karras, and A. Bernstein, "Hexastore: sextuple indexing for semantic web data management," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 1008–1019, 2008.

[13] K. Mulay and P. S. Kumar, "SPOVC: a scalable RDF store using horizontal partitioning and column oriented DBMS," in *Proceedings of the 4th International Workshop on Semantic Web Information Management (SWIM '12)*, ACM, May 2012.

[14] M. Atre, J. Srinivasan, and J. Hendler, "Bitmat: a main-memory bit matrix of RDF triples for conjunctive triple pattern queries," in *Proceedings of the International Conference on Posters and Demonstrations*, vol. 401, pp. 1–2, Aachen, Germany, 2007.

[15] A. Matono, S. M. Pahlevi, and I. Kojima, "RDFCube: a P2P-based three-dimensional index for structural joins on distributed triple stores," in *Databases, Information Systems, and Peer-to-Peer Computing: International Workshops, DBISP2P 2005/2006, Trondheim, Norway, August 28-29, 2005, Seoul, Korea, September 11, 2006, Revised Selected Papers*, vol. 4125 of *Lecture Notes in Computer Science*, pp. 323–330, Springer, Berlin, Germany, 2007.

[16] L. Zou, J. Mo, L. Chen et al., "gStore: answering SPARQL queries via subgraph matching," *Proceedings of the VLDB Endowment*, vol. 4, no. 8, pp. 482–493, 2011.

[17] P. Yuan, P. Liu, B. Wu et al., "TripleBit: a fast and compact system for large scale RDF data," *Proceedings of the VLDB Endowment*, vol. 6, no. 7, pp. 517–528, 2013.

[18] K. Kim, B. Moon, and H.-J. Kim, "R3F: RDF triple filtering method for efficient SPARQL query processing," *World Wide Web*, vol. 18, no. 2, pp. 317–357, 2013.

[19] B. Wu, Y. Zhou, P. Yuan, L. Liu, and H. Jin, "Scalable SPARQL querying using path partitioning," in *Proceedings of the IEEE 31st International Conference on Data Engineering (ICDE '15)*, pp. 795–806, IEEE, Seoul, South Korea, April 2015.

[20] O. Udrea, A. Pugliese, and V. S. Subrahmanian, "GRIN: a graph based RDF index," in *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI '07)*, vol. 1, pp. 1465–1470, July 2007.

[21] T. Tran and G. Ladwig, "Structure index for RDF data," in *Proceedings of the Workshop on Semantic Data Management (SemData@ VLDB '10)*, 2(010), 2010.

[22] H. He, H. Wang, J. Yang, and P. S. Yu, "BLINKS: ranked keyword searches on graphs," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '07)*, pp. 305–316, ACM, Beijing, China, June 2007.

[23] SPARQL, http://www.w3.org/TR/rdf-sparql-query/.

[24] U. Deppisch, "S-tree: a dynamic balanced signature index for office retrieval," in *Proceedings of the 9th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '86)*, pp. 77–87, San Diego, Calif, USA, October 1986.

[25] Y. Guo, Z. Pan, and J. Heflin, "LUBM: a benchmark for OWL knowledge base systems," *Web Semantics*, vol. 3, no. 2-3, pp. 158–182, 2005.

[26] M. Schmidt, T. Hornung, G. Lausen, and C. Pinkel, "SP2Bench: a SPARQL performance benchmark," in *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE '09)*, pp. 222–233, IEEE, Shanghai, China, April 2009.

[27] Uniprot RDF, http://www.ebi.ac.uk/uniprot/.

[28] M. Atre, V. Chaoji, M. J. Zaki, and J. A. Hendler, "Matrix 'bit' loaded: a scalable lightweight join query processor for RDF data," in *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, pp. 41–50, Raleigh, NC, USA, April 2010.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 4502867, 10 pages
http://dx.doi.org/10.1155/2016/4502867

Research Article

An Indoor Ultrasonic Positioning System Based on TOA for Internet of Things

Jian Li,[1,2] Guangjie Han,[1] Chunsheng Zhu,[3] and Guiqing Sun[4]

[1] *College of Internet of Things Engineering, Hohai University, Changzhou 213022, China*
[2] *State Key Laboratory of Acoustics, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China*
[3] *Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada V6T 1Z4*
[4] *Ocean College, Zhejiang University, Zhoushan 316021, China*

Correspondence should be addressed to Guangjie Han; hanguangjie@gmail.com

Received 8 September 2016; Accepted 1 November 2016

Academic Editor: Qingchen Zhang

With the development of Internet of Things, the position information of indoor objects becomes more important for most application scenarios. This paper presents an ultrasonic indoor positioning system, which can achieve centimeter-level precise positioning of objects moving indoors. Transmitting nodes, receiving nodes, and display control terminal are needed to constitute the entire system. The system is based on long-baseline positioning technology that uses code division multiplexing access mechanism. There is no limit to the number of receiving nodes as the system works in the up-transmit-down-receive mode. Positioning of a receiving node is found based on ultrasonic Time of Arrival ranging technology. To accurately determine the positioning, there must be at least four or five transmitting nodes. The working radius will not be less than 5 meters when the height is larger than 3 meters. The system uses wideband pseudorandom noise signal called Gold sequences for multiuser identification and slant range measurement. The paper first gives a brief introduction of popular indoor ultrasonic positioning methods and then describes the theory of proposed algorithm and provides the simulation results. To examine the correctness of the approach and its practicality, the practical implementation and experimental results are provided also in the paper.

1. Introduction

With the advent of Internet of Things (IoT) [1], most of the day-to-day objects (things) in the world will have computation and wireless communication capabilities with unique identity (ID) and Internet Protocol (IP) address. In most of the applications of IoT, we need to know the position (exactly or approximately) of the things joining the network [2, 3]. Getting the position information of things is highly important in many applications, and it is referred to as Location Based Service (LBS). In commercial applications, LBS describes a value-added service that provides an object's position information with the support of Geographic Information System (GIS) platform. The position information can also be acquired through mobile telecommunications operator's radio network or an external positioning method. LBS serves two purposes: first, to determine the location of a mobile device or user; secondly, to provide other information services related to location.

Global Positioning System (GPS) signal is typically unreachable in indoor environments, which makes it necessary to have an indoor positioning technology to accurately determine the position of an object indoors. The indoor positioning technology is used as an auxiliary or an alternative of GPS, where the GPS satellites' signal is weak as it reaches the earth but cannot penetrate a building.

Indoor positioning is highly useful in many applications, such as public safety and emergency response, positioning guide, social demand, market promotion demand, and large data applications. For public safety and emergency response applications such as fire disaster and rescue workers would be highly effective if victims of the disaster can be accurately localized, within the building to the granularity of a floor or a room number. In day-to-day life, LBS can help in identifying

Table 1: List of common indoor positioning methods.

Method	Accuracy	Complexity	Cost	Scope
UWB	Medium	Higher	Higher	lower
RFID	High	High	Low	High
ZigBee	Medium	Lower	Lower	Medium
Infrared	High	Low	High	Lower
Bluetooth	Low	High	Medium	High
WiFi	Lower	High	Lower	Higher
Ultrasonic	Higher	Lower	Lower	Medium

the location of a person's car in an underground parking lot or the location of milk in a supermarket. Indoor positioning can help in finding the nearest restaurant in a big shopping mall and the way to get there.

Indoor positioning refers to the process of determining the position of objects in an indoor environment. In many cases indoor positioning makes use of wireless communication, base station location, Inertial Navigation System, and a variety of similar techniques. Common indoor positioning technologies include [4–6] Wireless Fidelity (WiFi), Bluetooth, infrared, Radio Frequency (RF), Ultra Wide Band (UWB), Radio Frequency Identification (RFID), ZigBee, and Received Signal Strength (RSS) ultrasound. The authors in [4, 5] summarize these common indoor positioning methods and evaluate them with respect to positioning accuracy, coverage, cost, and complexity as shown in Table 1. These technologies are also important for the wireless sensor network research [7–9].

The technique proposed in [10] is based on a range-based positioning method using the physical layer of ZigBee. The authors studied a classic case which shows that the initial position accuracy plays an important role in accurate indoor positioning. Cotera et al. [11] applied trilateration algorithms that utilize radio frequency range estimation and their approach results in ±10 centimeters accuracy, with an overall of ±4.09 centimeters. The authors in [12] studied several Wireless Local Area Network (WLAN) indoor positioning methods based on RSS technology and studied appropriate selection criterion for grid size and Access Point (AP) reduction. Weighted Average Tracker (WAT) [13] method is also based on RSS where the accuracy is better than the traditional methods but is still of the order of one meter. The database sizes required both for the learning and for the estimation phases grow rapidly as the network coverage areas and the number of access points number increase. Spectral compression [14] approach has significantly reduced the database sizes for both the system learning and the estimation. Zheng et al. [15] proposed indoor 3D positioning system that utilizes low cost foot mounted MEMS (Micro Electro Mechanical System) sensors. In this approach, the range estimation deviation is about 1%, and the estimated coordinate errors are below one percent of the total transmission distance. Zhuang et al. [16] propose a two-filter integration approach for indoor positioning with MEMS sensors. The experiment results showed that the method has the accuracy of about several meters, despite the improvements in both the positioning accuracy and the computational efficiency.

In addition, to optimize the cover range and positioning accuracy, Domingo-Perez et al. [17] proposed an optimal sensor deployment method for indoor localization. Authors in [18] proposed a passive positioning method by adopting special characteristics of MIMO (Multiple Input Multiple Output) system where the target does not need to carry a positioning device. The system can reach accuracy about 1 meter. In a hospital environment, Haute et al. [19] evaluated a system with one anchor node in every two rooms. By adopting fingerprinting approach, the research points out that the positioning accuracy is about 1.21 meters and room determination accuracy is about 98%.

The above-mentioned positioning technologies take into account the interior structure of the indoor environment. The positioning accuracy between different methods has large variance. To achieve high precision in positioning, three ranging technologies such as ZigBee and ultrasonic are the most appropriate. Normally, ZigBee technology and RF technology keep the positioning accuracy up to meter scale, while ultrasonic technology can provide accuracy within centimeter range.

The rest of this paper is organized as follows: Section 2 gives a brief introduction of popular indoor ultrasonic positioning methods. The theory of proposed algorithm is described in Section 3 and the simulation results are provided in Section 4. To examine the correctness of the approach and its practicality, Section 5 describes the practical implementation and experimental results. Finally, Section 6 provides the conclusion and some remarks for future work.

2. Related Work

Compared with the nonacoustic methods discussed above, acoustic or ultrasonic methods can get higher accuracy in indoor positioning with lower cost. Ultrasonic positioning system is similar to radar and sonar systems, mainly including three parts: a transmission module, a transmission channel, and a receiver module. The frequency of the ultrasonic wave used in the indoor positioning system is mainly about 40 kHz [20], which can be narrow-band signal or wide band signal. Typical ultrasonic location systems mainly include: active bat method, cricket method, and dolphin method. Among these, the cricket method possesses lower cost. This method was developed by the Massachusetts Institute of Technology of United States.

Lindo et al. [21] introduce two multiband waveform synthesis methods for ultrasonic indoor positioning systems.

The horizontal error of their results is below 35 cm. With a portable grid of beacons and a few fixed anchors, De Angelis et al. [22] proposed an ultrasonic positioning system, which, in a system of 7 beacons, exhibits subcentimeter positioning accuracy in a range of up to 4 m. The approach proposed in [23] uses near-ultrasonic sound (17 kHz) with the errors less than about 2 cm in a noisy environment. The range of the positioning is within one meter square. To increase the accuracy of range estimation, the authors in paper [24] applying cross correlation technique use the receive signal and the reference stored in memory. Taking into account the possibility of loss of the ultrasonic positioning signal, based on Inertial Units measuring (IMUs), the literature [25] proposed a combined moving target location method, which utilizes extended Kalman filter (EKF) and Least Squares Support Vector Machine (LS-SVM). In the research of indoor positioning system, the energy strategy [26–28] is also important. This is mainly because the nodes used in the system can only be supplied by battery in most of the cases.

Ultrasonic indoor positioning system (IPS) can be classified using the following different criteria:

(i) Based on signal transmission and reception, IPS can be up-transmit-down-receive or up-receive-down-transmit;

(ii) Based on distance measurement techniques, IPS can be classified as Time of Arrival (TOA) systems and Time Difference of Arrival (TDOA) system;

(iii) Based on node synchronization, there are transceiver synchronization system and transceiver asynchronization system.

In general, ultrasonic positioning system not only has better positioning accuracy than radio frequency system but also has lower cost.

For the up-transmit-down-receive system, the ceiling nodes transmit ultrasonic signal, the ground nodes received ultrasonic signal, similar to the functioning of GPS positioning system; for the up-receive-down-transmit system, the ceiling nodes receive the ultrasonic signal transmitted by the ground node, similar to the Beidou satellite positioning system. TOA method needs to add the special time stamp into the transmit signal and use the signal arrival time to calculate the target space coordinates. On the other hand, TDOA method does not need the time stamp; it can directly calculate the space position of the target through the time difference of subsequent transmissions.

3. Theory of the System

Our proposed ultrasonic positioning system adopts TOA ranging technology to find the position of the receiving node. Our system consists of three types of nodes: control node, transmitter node, and receiver node, as shown in Figure 1.

In the ultrasonic location system, the most important nodes are anchor nodes and mobile nodes. Anchor nodes are generally fixed on the roof, while mobile nodes move with the target. When system is working, the transmit signals can be emitted from either the anchor node or the mobile node, corresponding to the receiver can be either the mobile node or the anchor node. The first situation is called up-transmit-down-receive mode and the second situation is called up-receive-down-transmit mode. Both the two modes can achieve the positioning function. The system mentioned here uses up-transmit-down-receive pattern. The transmitter nodes are mounted on the ceiling at preset positions, and they transmit ultrasonic signals. While the receiver node receives the ultrasonic signals from the transmitter nodes, and calculates the position itself. It then sends the position information to the console node through WiFi module.

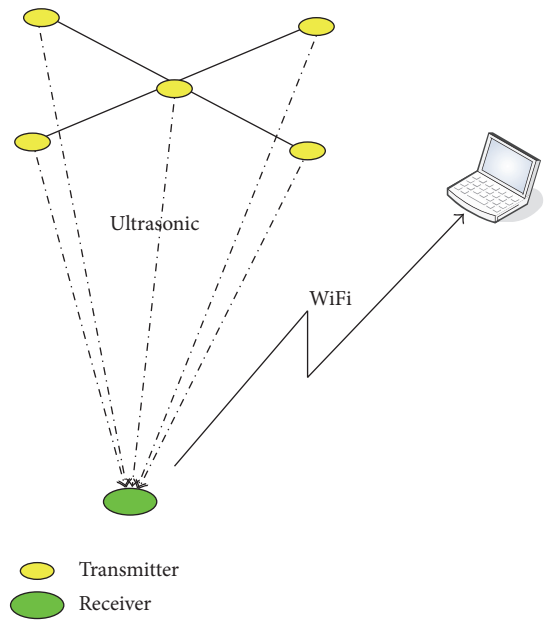


Figure 1: Positioning system structure diagram.

The positioning algorithm uses spherical intersection method to determine the coordinates of the mobile receiver node. Assume that the system has m transmitter nodes where the coordinates of each transmitter node are (x_n, y_n, z_n), where $n = 1, 2, 3, \ldots, m$. To calculate the coordinates of the receiver node, (x, y, z), we form the following spherical equation set (equation (1)) with the position and the slant distance between each transmitter and receiver node:

$$\begin{aligned}
(x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2 &= r_1^2 \\
(x_2 - x)^2 + (y_2 - y)^2 + (z_2 - z)^2 &= r_2^2 \\
(x_3 - x)^2 + (y_3 - y)^2 + (z_3 - z)^2 &= r_3^2 \\
&\vdots \\
(x_m - x)^2 + (y_m - y)^2 + (z_m - z)^2 &= r_m^2,
\end{aligned} \tag{1}$$

where $r_1, r_2, \ldots, r_m$ are the calculated distance with the TOA technique.

We can then estimate the position of the receiver node by using the least squares method, as shown in (2):

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} = \mathbf{A}^{-1}\mathbf{B}, \tag{2}$$

where $\hat{x}, \hat{y}, \hat{z}$ are the estimated values of (x, y, z) and the intermediate parameters $\mathbf{A}$ and $\mathbf{B}$ are as follows:

$$\mathbf{A} = 2\begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \\ \vdots & \vdots & \vdots \\ x_m - x_1 & y_m - y_1 & z_m - z_1 \end{bmatrix},$$
$$\mathbf{B} = \begin{bmatrix} d_2 - d_1 + r_1^2 - r_2^2 \\ d_3 - d_1 + r_1^2 - r_3^2 \\ d_4 - d_1 + r_1^2 - r_4^2 \\ \vdots \\ d_m - d_1 + r_1^2 - r_m^2 \end{bmatrix}, \tag{3}$$

where $d_n = x_n^2 + y_n^2 + z_n^2$ and $n = 1, 2, 3, 4, \ldots, m$.

In addition, the estimated value $\hat{z}$ should be given by the following equation (4), and the choice of positive and negative values should be according to the actual situation.

$$\hat{z} = \pm\sqrt{r_n^2 - (x_n - \hat{x})^2 - (y_n - \hat{y})^2} + z_n. \tag{4}$$

The accuracy of positioning is closely related to the geometry of the transmitter nodes and the receiver node. The error in positioning due to the geometry is called Position Dilution of Precision (PDOP). PDOP is a three-dimensional position precision factor. For a better positioning accuracy, the PDOP value should be small.

$$\mathrm{PDOP}^2 = \mathrm{HDOP}^2 + \mathrm{VDOP}^2, \tag{5}$$

where VDOP means Vertical Dilution of Precision and HDOP means Horizontal Dilution of Precision. HDOP is the square root of Latitude DOP (LaDOP) square plus longitude DOP (LoDOP) square:

$$\mathrm{HDOP}^2 = \mathrm{LaDOP}^2 + \mathrm{LoDOP}^2. \tag{6}$$

In up-transmit-down-receive system the number of receiver nodes is not restricted and the positioning operation is done by the receiver nodes, so the resource demand is more for receiver nodes compared to the transmitter nodes. The system is suitable for location and navigation integration services.

Velocity of ultrasonic waves is dependent on the ambient temperature. It is necessary to correct for the velocity of sound with respect to the temperature in order to ensure high precision in slant range measurement (centimeter to subcentimeter level).

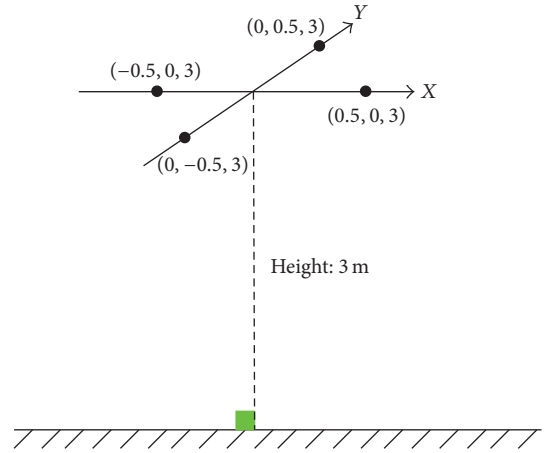


Figure 2: Nodes arrangement sketch.

$$C = 331.5 + 0.607t \ (\mathrm{m/s}), \tag{7}$$

where t is the environment temperature (°C); the sound speed is about 344 m/s at 20°C.

The console node can be any equipment (mobile phones or computers) which can access the WiFi network. After installing the indoor positioning software, user can view the spatial coordinate position of the moving object in real time and even realize the planning and management of the target trajectory.

4. Simulation

Multiple access to a medium can only be successful if the signals transmitted by different users are orthogonal to each other in the signal space: Frequency Division Multiple Access (FDMA) is orthogonal in the frequency domain; Time Division Multiple Access TDMA is orthogonal in the time domain; and Code Division Multiple Access CDMA is orthogonal in the users' characteristic waveforms. With orthogonal codes, the CDMA can achieve the multiuser positioning in the same time while using the same frequency band; the positioning efficiency is the highest. Gold sequence is the most common code used in CDMA system.

In our simulation and our implementation, the system uses wideband pseudorandom noise (PRN) signal, Gold sequence for multiuser identification (CDMA) and slant distance measurement. This helps in handling more users and improving the positioning accuracy.

Our typical application simulation scenario is shown in Figure 2: at least four transmitter nodes are installed on the ceiling, distributed evenly on the circumference of a circle of radius of 0.5 meters, and coordinates are (0, 0.5, 3), (0, −0.5, 3), and (0.5, 0, 3), (−0.5, 0, 3), respectively. The coordinates have a standard deviation (STD) of 1 cm, and the receiver node is at the coordinate (0.1, 0.2, 0).

The simulation results of 1000 times of Monte-Carlo experiments are shown in Figures 3 and 4. The red asterisk

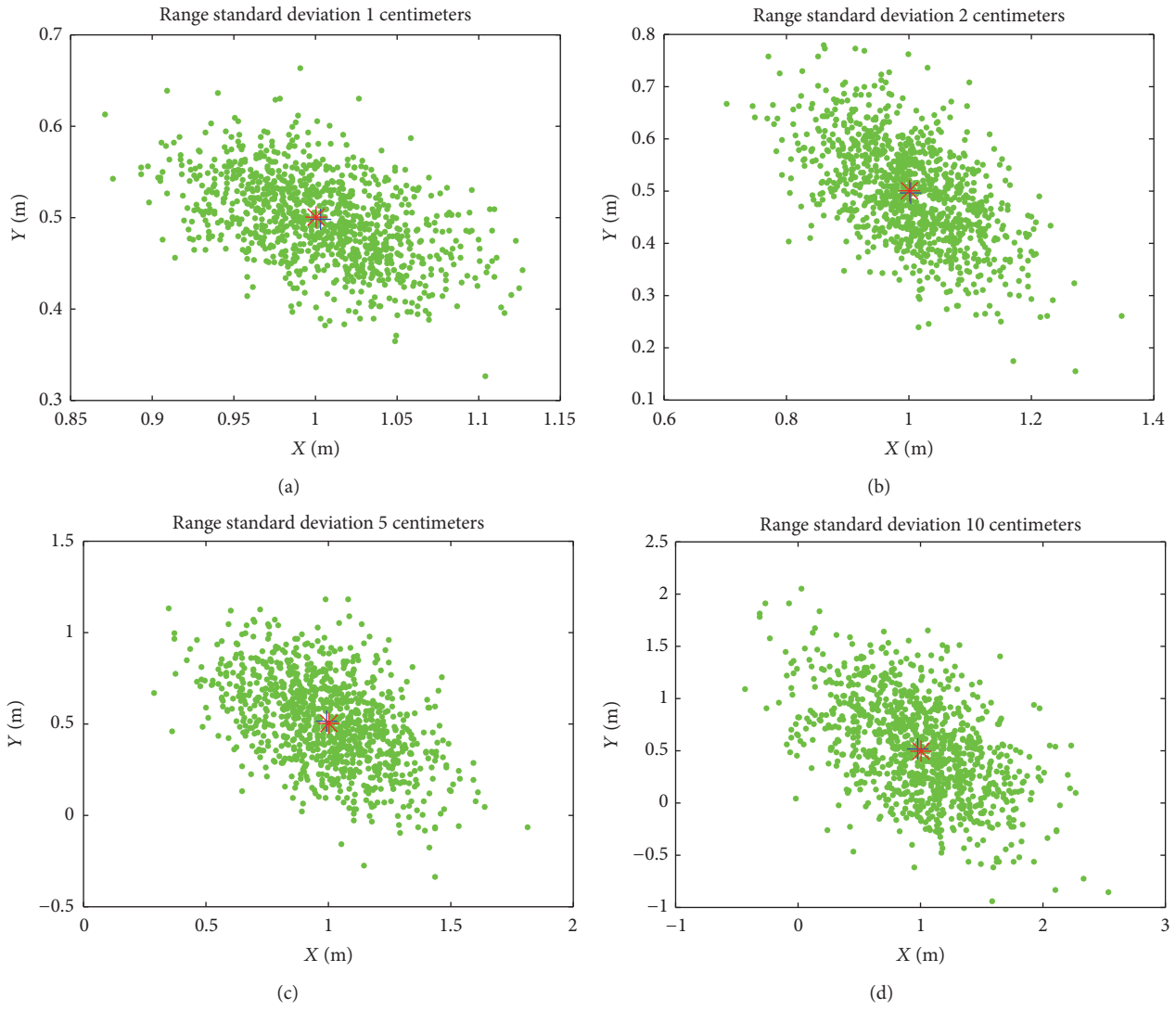


Figure 3: Target horizontal plane positioning result for different range STD.

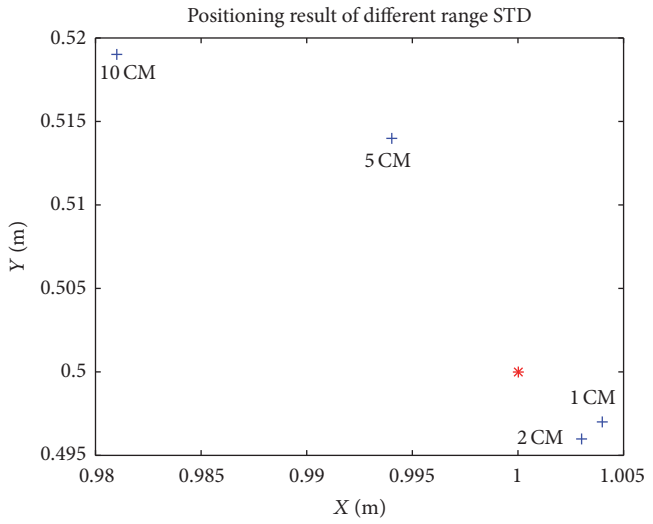


Figure 4: Positioning result of different range STD.

Table 2: Coordinate STD versus range estimate STD (cm).

Range estimate STD	1	2	5	10
X estimate STD	2.2	4.2	10.2	21.9
Y estimate STD	2.7	5.4	13.3	28.3

Table 3: Coordinate STD versus radius of transmit nodes (cm).

Radius of transmit nodes	400	200	100	50
X estimate STD	0.9	1.0	1.3	2.2
Y estimate STD	1.1	1.4	1.7	2.8

"$*$" in the plots marks the actual position of the receiver node, the green dots represent the results from 1000 simulations, and the blue plus sign represents the mean of the 1000 times' estimated position. The four simulation results correspond to the range estimation with standard deviations of 1 cm, 2 cm, 5 cm, and 10 cm.

It is evident from Figure 4 that with an increase in the range estimation standard deviation, the positioning error increases. When the range measurement standard deviation is in the order of one centimeter, a good positioning result can be achieved. In addition, the location error is large when the range error is large. As shown in Table 2, when the range estimate standard deviation is about 1 centimeter, statistics show that the X coordinates of the estimated standard deviation is about 2.2 cm and Y standard deviation is about 2.7 cm.

PDOP from the above simulations is 2.17, 4.12, 10.86, and 21.86 corresponding to range estimation standard deviations 1 cm, 2 cm, 5 cm, and 10 cm, respectively.

Above simulation results show that the positioning accuracy is dependent on range estimation. The following simulation takes into account the radius of the transmitter nodes. In this situation, we set the range estimation standard deviation at one centimeter and then observe the positioning accuracy with the radius of the transmit nodes at 50 cm, 100 cm, 200 cm, and 400 cm, respectively. The positioning results are shown in Figure 5, and we can see that the positioning standard deviation reduces with the increase in the radius of transmitter nodes. The positioning accuracy increases until the radius reaches four meters, but the coordinate STD still remains at about 1 centimeter regardless of any further increase in the transmitter node radius.

The coordinate standard deviation versus the radius of transmit nodes is shown in Table 3.

5. System Implementation

The system module block function setting, the transmitter and receiver circuit implementation, the transducer selection, and so on, all should be carefully considered to achieve high accuracy in an ultrasonic indoor positioning system.

The transmission signal has a decisive influence on the performance of the system. According to the transmitting signal ambiguity function theory, the measurement accuracy is mainly affected by signal frequency bandwidth and SNR (signal-to-noise ratio). Higher transmission signal bandwidth and higher received SNR typically help in achieving higher positioning accuracy. In fact, due to the limitation of ultrasonic transducer's transmitting response and the channel response, emission signal is limited to a narrow-band signal with bandwidth less than 4 kHz. Received SNR depends on the sound source level, transmission loss and the background noise level, and several other factors. SNR can be estimated by the sonar equation. In general, higher transmitter's sound level, smaller propagation loss, and lower background noise would result in higher SNR.

Figure 6 shows the block diagram of an ultrasonic signal transmitter node. The digital part includes three modules: (a) generation of the baseband signal, (b) coding pulse shaping, and (c) modulation of the coded signal. The resulting digital waveform can be stored in a register to be transmitted by the controller at regular time intervals. Having a signal amplifier and a band pass filter, before passing the digital waveform to a high quality analog converter, ensures better signal fidelity.

Receiver node is a key technology of the system, which is responsible for Doppler compensation, quadrature demodulation, and sliding correlation of the received signal. It also provides pseudorange measurement input for the positioning module. A block diagram of the receiver node is shown in Figure 7.

After several iterations of design and debugging, the transmitter nodes' and receiver nodes' circuits were made, as shown in Figure 8; Figure 8(a) is transmit node and Figure 8(b) is receiver node. This system has a center frequency at 40 kHz and bandwidth of about 4 kHz. The effective working distance can reach up to 10 meters. Receiver nodes are also equipped with a WiFi module to communicate with the console.

The system uses Gold sequence as transceiver signal to complete the CDMA encoding. Figure 9 shows the signals received by the receive node and the corresponding estimated Time of Arrival. In Figure 9(a), the vertical axis means the amplitude of the receive signal corresponds to the A/D transform output range from 0 to 4096, which means 0 refers to −3.3 V, 2048 refers to 0 V, and 4096 refers to +3.3 V, respectively. The horizontal axis corresponds to the sample points, which can be transferred to the time in accordance with the corresponding relationship. Figure 10 shows the positioning result of the system. When the Receiver node is stationery at (0.23, 0.13), we can see that the estimated standard deviation of both X and Y coordinates is less than 2 cm.

Both the simulation result and the field test result show that the method this paper proposed can provide high accuracy in case of indoor positioning. The positioning accuracy from our system is compared with other techniques as shown in Table 4.

6. Conclusion

Position information is important for most of the IoT applications. In this paper, an indoor positioning method is presented and the hardware and software are developed, which includes three kinds of basic nodes: transmitter

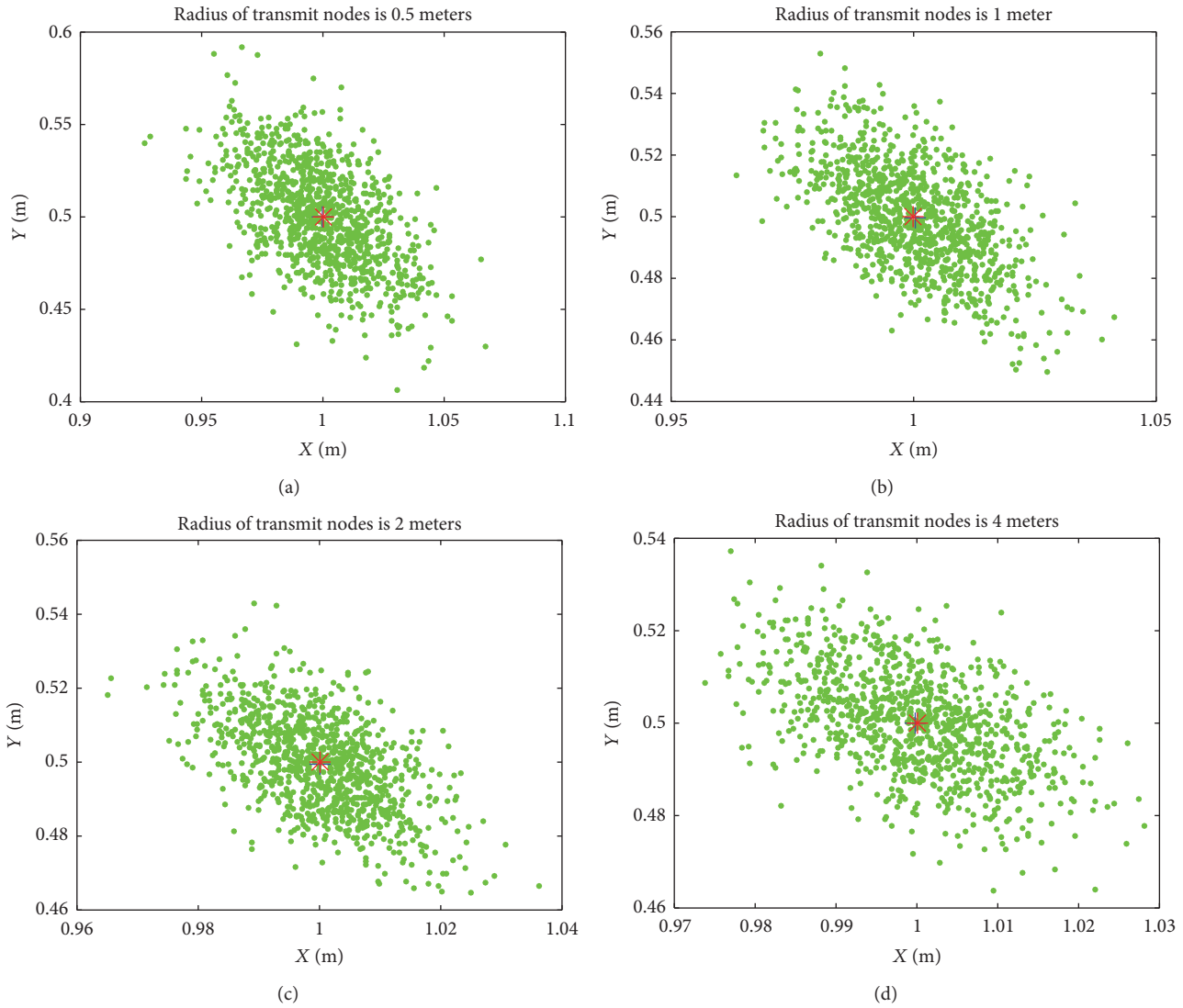


Figure 5: Positioning result for different radius at range STD of 1 cm.

Table 4: Comparison between this method and other technologies.

Method	Image	RFID	Bluetooth	WiFi	UWB	Infrared	ZigBee	Ultrasonic
Accuracy (meter)	10^{-6}~10^{-1}	10^{-2}~1	2~3	3~40	10^{-1}~1	10^{-2}~1	1~10	*2 cm*

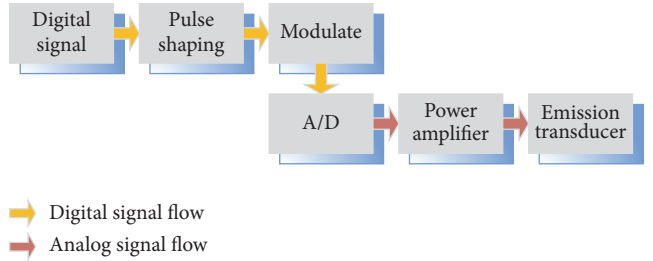


Figure 6: Block diagram of transmission node.

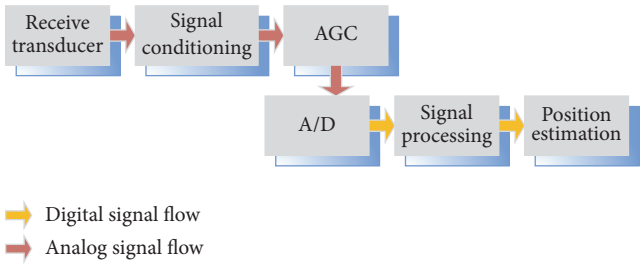
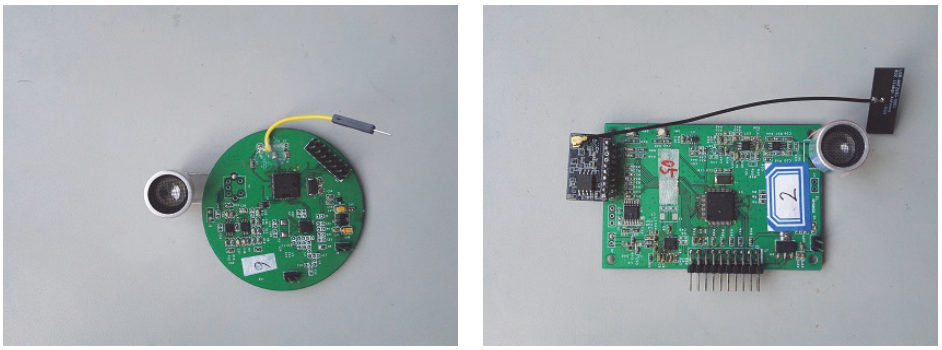


Figure 7: Block diagram of receive node.



(a) Transmitter node (b) Receiver node

Figure 8: Transmitter and receiver nodes.

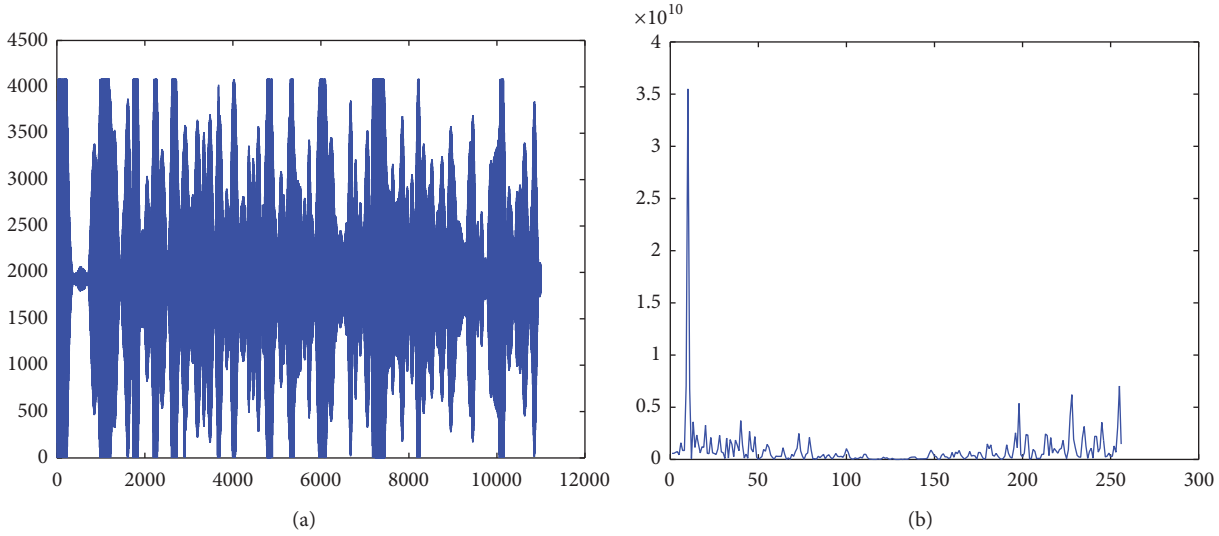


Figure 9: Receive time domain waveform and the TOA estimation.

nodes, receiver nodes, and console node. Transmitter nodes' spatial position are preset when mounting them onto the ceiling; receiver nodes receive the ultrasonic signal emitted by the transmit nodes and calculate its 3D coordinates. The console node displays the nodes' positioning results in real time. Our simulation setup and prototype system show that this method can achieve the accuracy of about several centimeters in an indoor positioning system.

Competing Interests

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

Acknowledgments

This work is sponsored by "Qing Lan Project," Natural Science Foundation of Jiangsu Province of China (no.

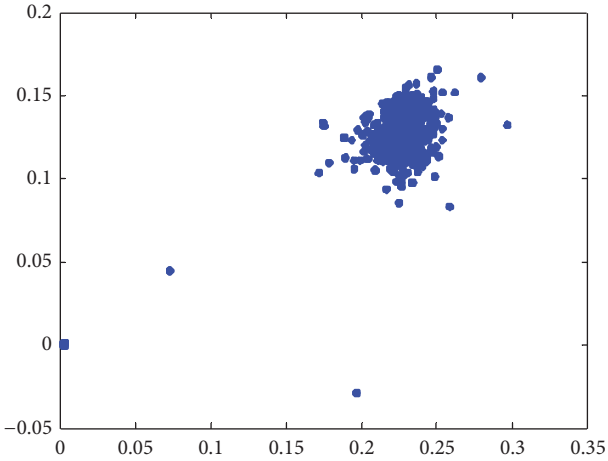


Figure 10: Target horizontal plane positioning result (experiment).

BK20161195), Open fund of State Key Laboratory of Acoustics (no. SKLA201504), National Natural Science Foundation of China (Grants nos. 61572172, 61571007, and 61273170), Fundamental Research Funds for the Central Universities (2013B18514, 2015B25214, and 2016B10714), "Changzhou Sciences and Technology Program, (nos. CE20165023 and CE20160014)," and "Six Talent Peaks Project in Jiangsu Province, no. XYDXXJS-007."

References

[1] C. Zhu, V. C. M. Leung, L. Shu, and E. C.-H. Ngai, "Green internet of things for smart world," *IEEE Access*, vol. 3, pp. 2151–2162, 2015.

[2] G. Han, J. Jiang, C. Zhang, T. Q. Duong, M. Guizani, and G. K. Karagiannidis, "A survey on mobile anchor node assisted localization in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2220–2243, 2016.

[3] C. Zhu, Z. Sheng, V. C. M. Leung, L. Shu, and L. T. Yang, "Toward offering more useful data reliably to mobile cloud from wireless sensor network," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 84–94, 2015.

[4] A. Alarifi, A. Al-Salman, M. Alsaleh et al., "Ultra wideband indoor positioning technologies: analysis and recent advances," *Sensors*, vol. 16, no. 5, article 707, 2016.

[5] L. Wei and K. Yong, "Analysis and research on indoor positioning technology," *Modern Navigation*, vol. 2, pp. 86–93, 2016.

[6] G. Han, J. Shen, L. Liu, and L. Shu, "BRTCO: a novel boundary recognition and tracking algorithm for continuous objects in wireless sensor networks," *IEEE Systems Journal*, 2016.

[7] C. Zhu, X. Li, V. C. M. Leung, X. Hu, and L. T. Yang, "Job scheduling for cloud computing integrated with wireless sensor network," in *Proceedings of the 2014 6th IEEE International Conference on Cloud Computing Technology and Science (CloudCom '14)*, pp. 62–69, Singapore, December 2014.

[8] C. Zhu, H. Nicanfar, V. C. M. Leung, and L. T. Yang, "An authenticated trust and reputation calculation and management system for cloud and sensor networks integration," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 118–131, 2015.

[9] C. Zhu, H. Wang, X. Liu, L. Shu, L. T. Yang, and V. C. M. Leung, "A novel sensory data processing framework to integrate sensor networks with mobile cloud," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1125–1136, 2016.

[10] J. Rapinski and S. Cellmer, "Analysis of range based indoor positioning techniques for personal communication networks," *Mobile Networks and Applications*, vol. 21, no. 3, pp. 539–549, 2016.

[11] P. Cotera, M. Velazquez, D. Cruz, L. Medina, and M. Bandala, "Indoor robot positioning using an enhanced trilateration algorithm," *International Journal of Advanced Robotic Systems*, vol. 13, no. 3, p. 110, 2016.

[12] E. Laitinen and E. S. Lohan, "On the choice of access point selection criterion and other position estimation characteristics for WLAN-based indoor positioning," *Sensors*, vol. 16, no. 5, p. 737, 2016.

[13] C. Huang and H. Manh, "RSS-based indoor positioning based on multi-dimensional kernel modeling and weighted average tracking," *IEEE Sensors Journal*, vol. 16, no. 9, pp. 3231–3245, 2016.

[14] J. Talvitie, M. Renfors, and E. S. Lohan, "Novel indoor positioning mechanism via spectral compression," *IEEE Communications Letters*, vol. 20, no. 2, pp. 352–355, 2016.

[15] L. Zheng, W. Zhou, W. Tang, X. Zheng, A. Peng, and H. Zheng, "A 3D indoor positioning system based on low-cost MEMS sensors," *Simulation Modelling Practice and Theory*, vol. 65, pp. 45–56, 2016.

[16] Y. Zhuang, Y. Li, L. Qi, H. Lan, J. Yang, and N. El-Sheimy, "A two-filter integration of MEMS sensors and WiFi fingerprinting for indoor positioning," *IEEE Sensors Journal*, vol. 16, no. 13, pp. 5125–5126, 2016.

[17] F. Domingo-Perez, J. L. Lazaro-Galilea, I. Bravo, A. Gardel, and D. Rodriguez, "Optimization of the coverage and accuracy of an indoor positioning system with a variable number of sensors," *Sensors*, vol. 16, no. 6, p. 934, 2016.

[18] Y. Zhang and H. Wang, "Research on indoor device-free passive localization algorithm based on multiple-input multiple-output system," *Journal of Xinjiang University*, vol. 33, no. 3, pp. 327–332, 2016.

[19] T. Haute, E. Poorter, P. Crombez et al., "Performance analysis of multiple Indoor Positioning Systems in a healthcare environment," *International Journal of Health Geographics*, vol. 15, article 7, 2016.

[20] A. Sanchez, A. D. Castro, S. Elvira, G. Glez-De-Rivera, and J. Garrido, "Autonomous indoor ultrasonic positioning system based on a low-cost conditioning circuit," *Measurement*, vol. 45, no. 3, pp. 276–283, 2012.

[21] A. Lindo, E. García, J. Ureña, M. del Carmen Pérez, and Á. Hernández, "Multiband waveform design for an ultrasonic indoor positioning system," *IEEE Sensors Journal*, vol. 15, no. 12, pp. 7190–7199, 2015.

[22] A. De Angelis, A. Moschitta, P. Carbone et al., "Design and characterization of a portable ultrasonic indoor 3-d positioning system," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 10, pp. 2616–2625, 2015.

[23] S. Murata, C. Yara, K. Kaneta, S. Ioroi, and H. Tanaka, "Accurate indoor positioning system using near-ultrasonic sound from a smartphone," in *Proceedings of the 8th International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST '14)*, pp. 13–18, Oxford, UK, September 2014.

[24] S. Mirshahi and O. Mas, "A novel distance measurement approach using shape matching in narrow-band ultrasonic system," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 400–405, 2015.

[25] X. Chen, Y. Xu, Q. Li, J. Tang, and C. Shen, "Improving ultrasonic-based seamless navigation for indoor mobile robots utilizing EKF and LS-SVM," *Measurement*, vol. 92, pp. 243–251, 2016.

[26] G. Han, L. Liu, J. Jiang, L. Shu, and G. Hancke, "Analysis of energy-efficient connected target coverage algorithms for industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, 2015.

[27] G. Han, Y. Dong, H. Guo, L. Shu, and D. Wu, "Cross-layer optimized routing in wireless sensor networks with duty cycle and energy harvesting," *Wireless Communications and Mobile Computing*, vol. 15, no. 16, pp. 1957–1981, 2015.

[28] C. Zhu, V. C. M. Leung, L. T. Yang, L. Shu, J. J. P. C. Rodrigues, and X. Li, "Trust assistance in Sensor-Cloud," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS '15)*, pp. 342–347, Hong Kong, May 2015.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 2398464, 11 pages
http://dx.doi.org/10.1155/2016/2398464

Research Article

Phase Clustering Based Modulation Classification Algorithm for PSK Signal over Wireless Environment

Qi An,[1] Zi-shu He,[1] Hui-yong Li,[1] and Yong-hua Li[2]

[1]*Electronic Engineering Department, University of Electronic Science and Technology of China, Chengdu, China*
[2]*Air Force Airborne Academy, Guilin, China*

Correspondence should be addressed to Qi An; sunnycaroline@163.com

Received 18 April 2016; Accepted 14 July 2016

Academic Editor: Laurence T. Yang

Promptitude and accuracy of signals' non-data-aided (NDA) identification is one of the key technology demands in noncooperative wireless communication network, especially in information monitoring and other electronic warfare. Based on this background, this paper proposes a new signal classifier for phase shift keying (PSK) signals. The periodicity of signal's phase is utilized as the assorted character, with which a fractional function is constituted for phase clustering. Classification and the modulation order of intercepted signals can be achieved through its Fast Fourier Transform (FFT) of the phase clustering function. Frequency offset is also considered for practical conditions. The accuracy of frequency offset estimation has a direct impact on its correction. Thus, a feasible solution is supplied. In this paper, an advanced estimator is proposed for estimating the frequency offset and balancing estimation accuracy and range under low signal-to-noise ratio (SNR) conditions. The influence on estimation range brought by the maximum correlation interval is removed through the differential operation of the autocorrelation of the normalized baseband signal raised to the power of Q. Then, a weighted summation is adopted for an effective frequency estimation. Details of equations and relevant simulations are subsequently presented. The estimator proposed can reach an estimation accuracy of 10^{-4} even when the SNR is as low as -15 dB. Analytical formulas are expressed, and the corresponding simulations illustrate that the classifier proposed is more efficient than its counterparts even at low SNRs.

1. Introduction

With the development of science and technology, wireless network has become the main media of information transmission in recent decades, and it plays an important role in the field of communication, field of military, and other fields. Satellite wireless network communication technology meets the requirements of time and place for information transmission, with its wide coverage, good broadcasting ability, and the unlimited character of different geographical conditions at any time or place. In the field of electronic surveillance and electronic countermeasures, multiple sensors can make each combat unit share their reconnaissance information through effective collaborative working systems and generate an overall situation with high precision and reliability via information fusion. Thus, the technology of cooperative reconnaissance network based on multiple sensors has become a hot issue in the field of electronic warfare.

As one of the main modulation methods in wireless communication networks, PSK, including multiple phase-shift-keyed (MPSK) and other converted forms such as $\pi/4$ differential quadrature phase-shift-keyed ($(\pi/4)$DQPSK), is also one of the most common carrier transmission modes in wireless digital communications. It has high spectrum utilization ratio and strong anti-interference ability; more importantly, it is also relatively simple in circuit implementations. Because of the remarkable spectrum character and multiple demodulation methods, $(\pi/4)$DQPSK is widely used in satellite communication networks and mobile communication systems. The nonbalanced quadrature phase-shift-keyed (UQPSK) is a modulation mode transferring two different types and rates of binary bit stream data, which is established by different power distribution of two

orthogonal components of the carrier. In recent years, it has been widely used in satellite digital communication networks or transmission and tracking systems between aircraft and grounded processing systems.

The present algorithms of PSK signals' modulation recognition are mostly based on decision theory and statistic pattern [1–3]. The former [1] is usually analyzed via the maximum likelihood function. A sufficient statistic for classification is obtained and simplified, and then a suitable threshold is chosen for comparing with this statistical parameter to achieve the modulation classification. Based on this general principle, some improved algorithms, such as Quasi-Log Likelihood Ratio [4, 5] (qLLR), Average Log Likelihood Function [6] (ALLF), Sequential Probability Ratio Test [7] (SPRT), and other improvements, were proposed by home and abroad scholars. However, these methods require a lot of known parameters, have a sensitive touch of symbols' synchrony and mode mismatch, and lead to a huge computation, which limit its own practical application heavily.

According to different statistical classification characteristics, the statistic pattern recognition methods can be divided into a number of branches, which are mainly based on instant information in time domain or other transforming domains, spectral correlation [8] (e.g., high order cumulant), constellations, chaos theory, and fractal theory, and other properties. Extracted from the instant information of the received signal in time and frequency domains, several parameters are adopted in Traditional Digitally Modulated Signal Recognition Algorithm [3] (DMRA). This method has a large correct recognition set, which makes it suitable for real-time data analysis. However, since each threshold is heavily dependent on SNR, DMRA cannot be effectively accomplished in practice, especially in the situations with low SNR. The algorithm derived from wavelet transform [9–11] can extract the signals' instant phase accurately. Yet it is only suitable for pulse shaping signals and deteriorates seriously for other kinds of signals. Based on spectral correlation, high order cumulant [12, 13] has a great property of anti-noise. However, it is limited to its exponentially increasing computation as the signals' modulation order is bigger-than-equal eight and cannot achieve online real-time data analysis, which limits its practical application heavily. Besides, the methods based on constellations [14, 15] and fractal theory [16] and any other methods are restricted to various extents to apply in practice.

Inspired by data mining and image processing, some novel algorithms are proposed in the recent one or two years in noncooperative communication. The approach based on clustering algorithms is a new trend in Automatic Modulation Classification (AMC) for digital modulations. An advanced method derived from K-means algorithm is proposed by Weber et al. [17] for Quadrature Amplitude Modulation (QAM) and PSK signals. In this paper, a novel utility function which indicates the best fitting constellation diagram is defined for the AMC decision. Simulations and measurements in a real monitoring environment demonstrate its effectiveness. Xu et al. [18] proposed a new method for phase clustering. Originated from mountain cluster algorithm [19], this technique can achieve multiple peaks in only one calculation process and avoid repeated peak cutting. On the other hand, it has about seven times computation time than high order cumulant due to its principle of repeated searching. Moreover, the author did a fault analysis at the condition of frequency offset existence. Both the algorithms above are formulated in the following section as objects for performance comparison.

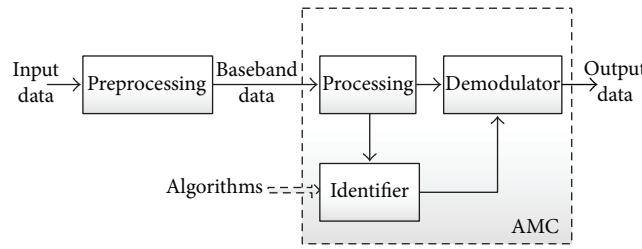


Figure 1: Receiver model.

In view of these above problems, this paper proposes an effective classifier. The structure is as follows: In the second part, one traditional and two novel methods of recent works on the PSK signal classification are formulated, and their shortcomings are pointed out, respectively. In the third part, an improved method is proposed and elaborated for classification. Furthermore, a robust estimator for frequency offset is presented and described in detail in the subsequent part. Necessary comparisons and simulations are performed and shown in the fifth part, which demonstrate the feasibility and effectiveness of proposed methods in this paper.

2. Recent Works

AMC is a significant step after signal detection in a radio monitoring environment and is fatal to the following process such as signals' demodulation and other steps. The simplified block diagram of the receiver is depicted in Figure 1. After series of preprocessing, the received signal in the receiver is transformed into a baseband signal. It is identified for signal modulation recognition, which leads to a more effective signal processing, such as demodulation and decoding, subsequently.

From the beginning of requirement in electronic monitoring and countermeasures, lots of researches are studied by home and aboard scholars. Their proposed algorithms and methods develop and improve the performance of NDA AMC of intercepted signals. A traditional algorithm, high order cumulant, and two novel methods, advanced K-means algorithm and phase clustering method, are introduced and elaborated in this section. They are adopted as the comparison objections.

2.1. High Order Cumulant. Assume that the signal to be processed $x(n)$ is a k order stationary random process with zero mean. According to the basic theory of stochastic processes, the k order cumulant and k order moment of $x(n)$ are both relative to time delay, yet irrelevant to the nth time spot.

For complex stationary random signal $X(t)$, its high order moment can be unitedly expressed as

$$M_{p+q,p} = E\left[X^p \left(X^*\right)^q\right], \tag{1}$$

where p and q are the index number of X and X^*, respectively. Several high order cumulants commonly used can be represented by high order moment as follows:

$$\begin{aligned} C_{20} &= M_{20}, \\ C_{21} &= M_{21}, \\ C_{40} &= M_{40} - 3M_{20}^2, \\ C_{41} &= M_{41} - 3M_{21}M_{20}, \\ C_{42} &= M_{42} - \left|M_{20}\right|^2 - 2M_{21}^2, \\ C_{60} &= M_{40} - 15M_4M_{20} - 30M_{20}^3. \end{aligned} \tag{2}$$

In theory, the algorithm of high order accumulation can completely eliminate the effect of Gaussian noise and be an ideal tool for signal processing under Gaussian noise. However, this method is not suitable for online real-time signal processing due to its amount of calculation. The higher modulation order the intercepted signal has, the more computation, which increases exponentially, it costs.

2.2. Advanced K-Means Method. The K-means algorithm is an optimal method for hard clustering when the number of clusters K in the input data x is known. Equation (3) is the cost formula, where i is the index and N is the length of the input data x:

$$J = \sum_{i=1}^{N} \left|x_i - z_{ik}P_K\right|^2. \tag{3}$$

The variable z_{ik} is the membership indicator, which is equal to unity if the input data x_i belongs to the cluster k and zero otherwise. The membership indicator z_{ik} is calculated based on the shortest distance of the input data x_i to the prototypes P_K, as given in (4), where w_i represents the index k of the winning prototype P_k for the input data x_i:

$$\omega_i = \arg\min_{P_K} \left(\left|x_i - P_K\right|\right). \tag{4}$$

The K-means algorithm iteratively solves the clustering problem stated in (4) by alternating between a competitive and a learning step. In the competitive step, the allocation of the input symbols x_i to the prototypes P_K is carried out in such a way that J in (3) is minimised. In the learning step, the prototype positions are updated by calculating the mean value of the corresponding input symbols x_i.

A novel utility function F is proposed by Weber that indicates the best fitting constellation diagram to the calculated prototypes of the clustering algorithm:

$$\begin{aligned} F &= F_1^2\left(C_{S,K}\right) * F_2\left(C_{S,K}\right), \\ F_1\left(C_{S,K}\right) &= \frac{1}{1 + J_{C_{S,K}}K}, \\ F_2\left(C_{S,K}\right) &= \frac{1}{1 + (1/K)\sum_{k=1}^{K}\left|C_k - P_k\right|} \frac{\sum_{k=1}^{K}\Phi\left(P_k\right)}{K}, \\ \Phi\left(P_k\right) &= \begin{cases} 1, & \text{for } \sum_{i=1}^{N} z_{ik} > 0 \\ 0, & \text{else,} \end{cases} \end{aligned} \tag{5}$$

where $J_{C_{S,K}}$ is the result of the cost function in (3) for a specific constellation pattern $C_{S,K}$ of the considered modulation pool $M_{S,K}$. The variable K represents the number of prototypes or the modulation order of $C_{S,K}$ and S represents the modulation scheme. C_K are the specific symbol positions of the constellation pattern. $C_{S,K}$ and P_K are the calculated prototypes.

Generally, the first term of the utility function $F_1(C_{S,K})$ indicates the minimisation of cluster variances, and the second factor $F_2(C_{S,K})$ evaluates the position of the calculated prototypes P_K to the given constellation pattern $C_{S,K}$ and the assignment of the input symbols x_i to the prototypes. To conclude, the utility function $F = F_1^2(C_{S,K}) * F_2(C_{S,K})$ evaluates if all prototypes are covered by the input samples and if the variance of the clusters and the EVM can be minimised. For this reason, this algorithm is named the highest constellation pattern matching (HCPM) algorithm.

A real monitoring environment is employed for field trial in Weber's paper. Eight signals including 4 kinds of QAM and MPSK signals, respectively, are introduced for identification capability demonstration. Two-Threshold Sequential Algorithmic Scheme (TTSAS) [20], fuzzy algorithm [20] and the K-centre algorithm, are adopted for performance comparison. Simulations and measurements show the effectiveness of the proposed method than the other three counterparts.

The idea gives a new direction for signal recognition. What is better, it can be extended to the application of QAM signal detection and identification. However, the simulation running time it needed is too long; that is to say, it is not relatively suitable for real-time signal processing than some other methods. The simulation time of this method is displayed in the table in Simulation, which shows that it has a larger weakness as it compares to its counterparts.

2.3. Phase Clustering Method. Another method based on data mining is phase clustering method proposed by Xu et al. [18]. Inspired by subtractive clustering method, a novel clustering function is derived for signals' classification. Because the method proposed in this paper is an improvement of this method, the specific process is no longer described here.

The carrier frequency offset is also considered in Xu's paper. The premise is that the received signal has been timing synchronized. When the carrier frequency offset Δf satisfies

$\Delta f T_b \leq 0.15$, an approximate exact complex sequence can be obtained from [21], and the phase sequence can be expressed as

$$\varphi_r(k) = \varphi_x(k) + 2\pi k \Delta f T_b + \varphi_n(k), \quad (6)$$

where T_b is the symbol period and $k = 1, \ldots, N$ is the sample spot, where N is the sample number of each data package. $\varphi_n(k)$ is the noise's phase, and the phase of received signal $\varphi_x(k)$ is

$$\varphi_x(k) \in \left\{ \frac{2\pi m}{M} + \theta_0,\ m = 0, 1, \ldots, M-1 \right\}. \quad (7)$$

In order to eliminate the influence of frequency offset as much as possible, a new sequence can be obtained by making difference to $\varphi_x(k)$; that is,

$$\begin{aligned} \varphi_r'(k) &= \varphi_r(k+1) - \varphi_r(k) \\ &= \varphi_x(k+1) - \varphi_x(k) + 2\pi \Delta f T_b + \varphi_n(k+1) \\ &\quad - \varphi_n(k) = \varphi_x'(k) + \varphi_n'(k), \end{aligned} \quad (8)$$

where

$$\varphi_x'(k) \in \left\{ \frac{2\pi m}{M} + 2\pi \Delta f T_b,\ m = 0, 1, \ldots, M-1 \right\}. \quad (9)$$

The major idea of Xu's method for reducing the influence of frequency offset is to employ the difference of phase of the baseband signal $\varphi_r'(k)$ as the signal to be processed, the same as the condition without frequency offset to get the correct signal modulation order. A significant premise mentioned in his paper is that $\varphi_x'(k)$ is a uniform distribution object. However, in this case, the effective part (9) is no longer subject to uniform distribution, which directly leads to the phase clustering algorithm invalid.

Four common MPSK signals are introduced for classification performance in XU's paper. The correction classification probability is adopted as the measure index.

In order to enhance the recognition efficiency, this paper proposes an improved method for phase clustering, which can effectively reduce the signal processing time without degrading the classification performance. In addition, in view of the carrier frequency offset, this paper also gives a feasible solution for its estimation and correction.

3. Proposed Method

In order to achieve a better NDA classification performance for signals, an improved phase clustering method is proposed in this paper. Then, a robust estimation method is proposed for frequency offset correction.

3.1. Advanced Phase Clustering Method. The received wave of modulated PSK signal can be generally expressed as

$$r(t) = \sum_n a_n g\left(t - nT_s\right) e^{j(2\pi f_c t + \theta_0)} + n(t), \quad (10)$$

where $a_n \in \{\exp(-j(2\pi m/M)),\ m = 0, 1, \ldots, M-1\}$ is the symbol sequence, M is the modulation order, $g(t)$ is the shaping pulse of shaping filter, T_s is the sample period, f_c is the carrier frequency, and θ_0 is the carrier phase (Figure 3). $n(t)$ is the white Gaussian noise with zero mean and N_0 variance.

Root Raised Cosine (RRC) filter is generally adopted for signal shaping in wireless communication networks. It is also considered in this paper. Under the premise of carrier and timing synchronization, the output baseband signal's phase after matching filter can be written as

$$\varphi_r(k) = \varphi_x(k) + \theta_0 + \varphi_n(k), \quad k = 1, 2, \ldots, N, \quad (11)$$

where $\varphi_x(k)$ denotes the phase sequence of transmitted signal, $\varphi_n(k)$ denotes the phase sequence of noise, and N is the sample number of each data package. For PSK signal, its phase $\varphi_x(k)$, expressed as follows, commonly obeys to uniform distribution, which means that there are M/N sample spots intensely around each constellation point if M-order modulated signal is sampled at N points:

$$\varphi_x(k) \in \left\{ \frac{2\pi m}{M},\ m = 0, 1, \ldots, M-1 \right\}. \quad (12)$$

In order to measure the radian distance between the phase of sampled point and reference phase, a distance function $D(\varepsilon)$ is defined with independent variable $\varepsilon \in [0, 2\pi)$, here,

$$D(\varepsilon) = \begin{cases} |\varepsilon| & |\varepsilon| \leq \pi \\ 2\pi - |\varepsilon| & |\varepsilon| > \pi. \end{cases} \quad (13)$$

An advanced clustering function is proposed for phase clustering and expressed as fractional form, which has a remarkable reduction on computation as comparing to the index form, as

$$v_r(\theta) = \sum_{k=1}^{N} \frac{1}{1 + (N/\pi) D\left(\varphi_r(k) - \theta\right)}, \quad (14)$$

where $\theta \in [0, 2\pi)$ denotes the phase variable as the reference phase.

A division set of θ is installed as the reference phase for phase clustering. As is shown in formula (14), all the phase to be processed need to measure the distance to each reference phase θ before clustering. In fact, the clustering process is similar to a repeated search process. Just because of this, the computational quantity of phase clustering is totally determined by the division set of θ. A suitable division can not only reduce the calculation amount of this method, but also enhance the correction rate of clustering. The uniform distance from 0 to 2π is a common method of reference phase segmentation.

For simplicity, an approximation is used to reduce calculation amount. When $D(\varphi_r(k) - \theta) \geq 9\pi/N$,

$$\frac{1}{1 + (N/\pi) D\left(\varphi_r(k) - \theta\right)} \leq 0.1. \quad (15)$$

And it can be regarded as

$$\frac{1}{1 + (N/\pi) D\left(\varphi_r(k) - \theta\right)} \approx 0. \quad (16)$$

When a baseband PSK signal is to be processed, the advanced phase clustering (APC) function can be expressed as

$$v_r(\theta) = \sum_{k=1}^{N} \frac{1}{1 + (N/\pi) D\left(\varphi_x(k) + \theta_0 + \varphi_n(k) - \theta\right)}. \quad (17)$$

Assume that $\varphi_x(k)$ is distributed independently and has an equal occurrence probability. Since $\varphi_n(k)$ is a stationary Gaussian phase noise, formula (17) can transform into

$$v_r(\theta) \approx \frac{N}{M} \sum_{m=0}^{M-1} \frac{1}{1 + (N/\pi) D\left(2\pi m/M + \theta_0 + \varphi_n'(m) - \theta\right)}. \quad (18)$$

Note that the necessary condition of the upper equation is the uniform distribution of the transmitted signal phase without noise. In the ideal constellation, there are approximate N/M sample points distributed on the location of each constellation, if the sample number of the received signal, which has M modulation order, is N. Otherwise, the above equation is not established anymore.

Due to the particularity of the distance function, $D(\varepsilon)$ has a periodicity of 2π. On the other hand, the independent variable θ in the clustering function (14) is only related to the distance function, which leads to the fact that the clustering function also has the periodicity of 2π.

The character of periodicity of formula (14) is given in detail. When $\theta + 2\pi/M < 2\pi$, the periodicity of clustering function is formulated and elaborated in formula (19). It also can be proved as the same way that $v_r(\theta + 2\pi/M - 2\pi) = v_r(\theta)$, if $\theta + 2\pi/M > 2\pi$:

$$\begin{aligned} v_r\left(\theta + \frac{2\pi}{M}\right) &= \frac{N}{M} \cdot \sum_{m=0}^{M-1} \frac{1}{1 + (N/\pi) D\left(2\pi m/M + \phi - (\theta + 2\pi/M)\right)} \\ &= \frac{N}{M} \left(\frac{1}{1 + (N/\pi) D\left(\phi - (\theta + 2\pi/M)\right)} \right. \\ &\quad + \sum_{m=2}^{M-1} \frac{1}{1 + (N/\pi) D\left(2\pi m/M + \phi - (\theta + 2\pi/M)\right)} \\ &\quad \left. + \frac{1}{1 + (N/\pi) D\left(2\pi/M + \phi - (\theta + 2\pi/M)\right)} \right) \\ &= \frac{N}{M} \left(\frac{1}{1 + (N/\pi) D\left(2\pi + \phi - (\theta + 2\pi/M)\right)} \right. \\ &\quad + \sum_{m=1}^{M-2} \frac{1}{1 + (N/\pi) D\left(2\pi m/M + \phi - \theta\right)} \\ &\quad \left. + \frac{1}{1 + (N/\pi) D\left(\phi - \theta\right)} \right) = \frac{N}{M} \\ &\quad \cdot \sum_{m=0}^{M-1} \frac{1}{1 + (N/\pi) D\left(2\pi m/M + \phi - \theta\right)} = v_r(\theta). \end{aligned} \quad (19)$$

The expression of clustering function $v_r(\theta)$ is a periodical function with

$$T_v = \frac{2\pi}{M}. \quad (20)$$

Since the periodic signal has special spectral properties in frequency domain, the period of clustering function can be extracted easily through Fast Fourier Transform (FFT), $V_\theta(\omega) = \mathrm{FFT}[v_r(\theta)]$. The certain frequency which is corresponding to the place of maximum value of its Fourier transform result indicates T_v. Modulation order M can be calculated through the above equation; then signals' classification is achieved. More favorably, carrier phase θ_0 is irrelevant to this method for T_v, which means that this proposed method is also robust to signal's constellation rotation.

If deep recognition requirements are needed for PSK signals with the same modulation order, lots of statistics parameters can be chosen and employed. For example, there are two sets of data: 4PSK and OQPSK, 8PSK and $(\pi/4)$DQPSK. Envelope entropy of differential phase of the baseband signals can be introduced to distinguish them, respectively. The recognition performance is certainly determined by the introduced statistics parameters, yet regardless of the APC method.

The APC method has several advantages:

(1) Since it is derived from the coding characters rather than statistical properties, the direct influence of SNR is reduced.

(2) As an optimization algorithm of multiple peaks searching, it achieves all the peaks in one calculation process and avoids repeated peak cuttings.

(3) Fraction is used instead of exponential function in the clustering function, so that the calculation process is simplified, which leads to a much lower computational quantity.

3.2. Frequency Offset Correction. In the digital communication system, the carrier frequency offset is often introduced by the difference between the receiver and the transmitter oscillator and also caused by the Doppler frequency shift, which is brought by the channel nonlinearity and phase noise. In the wireless network, especially the electronic monitoring and other noncooperative communication systems, the accuracy of frequency offset estimation directly affects the performance of the receiver.

In Xu's method mentioned in last section, phase clustering algorithm is directly adopted with the difference phase of received signal. However, the difference phase of the effective part of signal which carries messages is no longer uniform distribution. Only frequency offset estimation and correction can be considered in this case to eliminate the impact of frequency offset as much as possible.

The frequency offset estimations of the Fitz and L&R algorithms are directly achieved via the weighted summation of the autocorrelation of the signal. This means that these two algorithms, and the improved methods based upon them, are all heavily affected by the correlation interval on the estimated range, unless the correlation interval achieves its maximum value of sampled number N. However, in this case the calculation amount increases dramatically, especially when N is large. For this particular reason, most of the improved methods are derived from Kay's algorithm. The autocorrelation function of the processing signal is used in the M&M algorithm used in Kay's and L&W's algorithms for weakening the influence brought on by phase noise. However, the addition operation of the autocorrelation's phase introduces the phase folding problem. An objection phase of the baseband signal, which has a real value near $-\pi$ or π, may be changed to a completely different value under the influence of noise, and this leads to an error in the frequency offset estimation result. The WNALP algorithm is derived from the M&M algorithm, which solves the phase folding problem and broadens the estimation range remarkably. However, the signal in real noncooperative environments is usually intercepted under a low SNR due to its special condition, which causes great difficulties in subsequent signal processing.

In order to reduce the thresholds' effect and improve the unbalance between estimation accuracy and estimation range of frequency offset under low SNR, an advanced NDA estimator based on the weighted summation of the differential phase of the autocorrelation is proposed in this paper.

Assume that timing synchronization is accomplished. The baseband signal sequence with frequency offset $x(k)$ is expressed as

$$x(k) = c_k \exp^{j(2\pi f_\Delta k T_s + \theta)} + n(k), \tag{21}$$

where c_k $(k = 1, 2, \ldots, N)$ is the modulated symbol sequence from the transmitting end, N is the number of sampling points of the selected signal segment in the receiving end, f_Δ is the unknown frequency offset to be estimated, T_s is the sample period, and θ is a random initial carrier phase, which follows the uniform distribution in the range of $[0, 2\pi)$. Usually, the channel noise of the communication system $n(k)$ is considered to be random complex additive Gaussian noise, with zero mean and bilateral spectral density $N_0/2$. We normalize its amplitude as

$$\widetilde{x}(k) = \frac{x(k)}{|x(k)|}, \quad k = 1, 2, \ldots, N. \tag{22}$$

Then, under the hypothesis of $n(k) \gg 1$ for a large enough SNR,

$$\widetilde{x}(k) \simeq e^{j(2\pi k f_\Delta T_s + \theta + \widetilde{\beta}(k))}, \tag{23}$$

where $\widetilde{\beta}(k)$ is also a Gaussian process with zero mean.

The autocorrelation is defined as

$$R_0(m) = \sum_{k=m+1}^{N} \widetilde{z}(k)\, \widetilde{z}^*(k-m), \quad m = 1, 2, \ldots, L_r, \tag{24}$$

where $\widetilde{z}(k) = \widetilde{x}^Q(k)$ is the normalized baseband signal raised to the power of Q and L_r is the set maximum correlation interval. Using the same principle as above, the autocorrelation function can be continuously transformed as

$$R_0(m) \simeq e^{j(2\pi m f_\Delta T_s + \widetilde{\epsilon}(m))}, \tag{25}$$

where $\widetilde{\epsilon}(m)$ is also a Gaussian process with zero mean. We see that

$$\angle R_0(m)\, R_0^*(m-1) = 2\pi f_\Delta T_s + \widetilde{\epsilon}(m) + \widetilde{\epsilon}(m-1), \quad 2 \le m \le L_r. \tag{26}$$

We define $\Delta\varphi_{0R(m)} \triangleq \angle R_0(m) R_0^*(m-1)$. According to the principle of Kay's algorithm, an objective function can be set as

$$\mathbf{J}_0 = \left(\Delta\boldsymbol{\varphi}_\mathbf{0} - 2\pi f_\Delta T_s \mathbf{e}\right)^T \mathbf{C}^{-1} \left(\Delta\boldsymbol{\varphi}_\mathbf{0} - 2\pi f_\Delta T_s \mathbf{e}\right), \tag{27}$$

where $\Delta\boldsymbol{\varphi}_\mathbf{0} = [\Delta\varphi_{0R(1)}, \Delta\varphi_{0R(2)}, \ldots, \Delta\varphi_{0R(L_r)}]^T$. The estimated value of the frequency offset $\widehat{f}_\Delta$ is obtained when the objective function $\mathbf{J}_0$ obtains its minimum value. So, the normalized weighted correlation linear estimator proposed can be expressed as

$$\widehat{f}_\Delta = \frac{1}{2\pi Q T_s} \cdot \sum_{m=2}^{L_r} \omega_0(m) \cdot \arg\left[R_0(m)\, R_0^*(m-1)\right], \tag{28}$$

where ω_0 is the weight of the differential phase:

$$\omega_0(m) = \frac{3\left[(N-m)(N-m+1) - L_r\left(N - L_r\right)\right]}{L_r\left(4L_r^2 - 6L_r N + 3N^2 - 1\right)}. \tag{29}$$

The normalized baseband signal is considered to be the signal to be processed, which effectively weakens the performance loss resulting from the nonlinear operation of raising to a power of Q. The weighted summation of the differential phase of the autocorrelation also decreases the influence of noise effectively compared to the method of argument operation after weighting the conjugate difference of the autocorrelation. Thus it provides a better estimation accuracy and is described in Kay's paper [5]. The difference of autocorrelation is a great improvement, which can make the estimation range independent of the maximum correlation interval L_r and solve the phase folding problem compared to Fitz's and the L&R algorithm and the improved methods based upon them. Meanwhile, the proposed method in this

Table 1: Simulation time for order classification.

	Simulation time (s)
HOM method	0.0158
AKMC method	0.4523
PC method	0.0720
APC method	0.0345

paper has the same estimation range as its counterpart WNALP algorithm. Importantly, the large estimation variance of the WNALP algorithm under low SNR conditions is improved, which effectively balances the trade-off between estimation accuracy and estimation range under low SNR in the process of frequency offset estimation.

4. Simulation

Computer simulations are performed to test the performance of the methods proposed in this paper. Considering the background of electronic monitoring and countermeasure in satellite communication and wireless communication networks, the simulation set contains six common modulation types of PSK signals: BPSK, QPSK, 8PSK, 16PSK, OQPSK, $(\pi/4)$DQPSK, and UQPSK. Each simulation result is the average of 1000 independent runs. Because of the special environment we assumed, no *a priori* knowledge of intercepted signal is assumed for all the experiments.

Regular communication equipment and environment are adopted for these simulations. The signals are shaped by raised root cosine filter with its roll-off factor $\alpha = 0.22$. The received intercepted signal is sampled $N = 512$ points for test with Sample frequency $f_s = 20$ MHz. We suppose that the symbol rate N_s has been estimated accurately by a certain algorithm and the sample number in per symbol period is $N_b = N_s * N/f_s = 32$. Additive white Gaussian noise is considered in this situation. Moreover, channel effects such as fading and multipath propagation are ignored and we assume that perfect time and frequency synchronization have been achieved. The SNR in this paper is defined as E_s/N_0, where E_s is the energy per symbol and N_0 is the power spectral density of the Gaussian noise.

A common method based on four-order cumulant (HOC) and two new ways derived from data mining, advanced K-means clustering (AKMC) and phase clustering (PC), are adopted for performance comparison of signal classification. Classification capabilities and simulation times in a single run of the subroutines are shown in Figure 2 and Table 1, respectively.

The same set of signal data to be processed are introduced in four subroutines, respectively, for classification performance and simulation time comparison. Except for four-order cumulant, the other three methods present obvious correct classification rate trends and a large SNR tolerance for all the involved signals. The 16PSK signal can be correctly classified from approximate 2 dB, and the other six signals have larger SNR tolerances less than −5 dB. Even so, they are distinguished clearly via simulation time table. It is shown in Table 1 that PC method and AKMC method both cost two or more times simulation time than APC method proposed in this paper. If the modulation order of signal to be processed is 16, the order needs to be chosen as 16 for accumulation in HOC algorithm. However, the computation amount of this algorithm increases exponentially. It is a predictable result that its simulation time could be bigger than or equal to the time of APC method. Due to the instability of the APC algorithm at the low SNRs, few wrong judgements of the signal's modulation order appear. That is the reason why the order of 16PSK signal gets 17 during 1~2 dB.

In summary, APC method proposed in this paper has a better classification capability than its counterparts and gives a new guidance for practical signal processing. Order classification result and correct classification rate by APC method are separately displayed in Figures 4-5.

Frequency offset directly impacts on the performance of signal classification. In order to verify the effectiveness of the proposed NWALP method, the WNALP algorithm, from which the proposed method in this paper is derived, was selected for comparison in this paper.

Let us assume that the signal to be processed is a QPSK signal with additive Gaussian noise. The number of sample points is set as $N = 256$, and the sampling frequency is normalized to the unit as $f_s = 1$. The correlation interval of autocorrelation is set as $L_r = N/2$. Each simulation of estimation accuracy and estimation range for the frequency offset was run at least 100 times. The estimation variance is adopted as the measure of estimation accuracy. The MCRLB [22] is also calculated as an absolute measure of the theoretical optimal valuation:

$$\text{MCRLB} = \frac{3}{2\pi^2 N\left(N^2-1\right)} \cdot \frac{1}{E_s/N_0}. \tag{30}$$

The performance of the proposed method compared to the abovementioned methods is shown in Figure 5 under a normalized frequency offset $f_\Delta = 0.001$, as the SNR changes within the range of −15~20 dB by 1 dB steps. It can be seen that even when the frequency offset is set at a smaller value, the WNALP algorithm still shows a poor estimation performance of about 10^{-2} under low SNRs. This means that when the frequency offset is small, the error of the algorithm may be of the same order of magnitude as the frequency offset itself. Such a large estimation error leads to a complete failure of the algorithm. However, it can be obviously seen that the estimation accuracy of the proposed method remains steady in the vicinity of 10^{-4} under low SNR conditions and has an improved estimation accuracy of at least two orders of magnitude compared to the original WNALP algorithm. On the other hand, the estimation error of the proposed method rapidly decreases to a magnitude near the MCRLB. Even as the SNR increases, it remains steady at approximately 10^{-7} due to the SNR threshold effect.

The estimation range of the carrier frequency offset is usually observed under large SNRs in order to obtain wider and more accurate bounds. When SNR = 15 dB, the estimated ranges of the chosen algorithms were simulated as shown in Figure 6. As can be seen, the proposed NWALP method can achieve the same frequency offset estimation

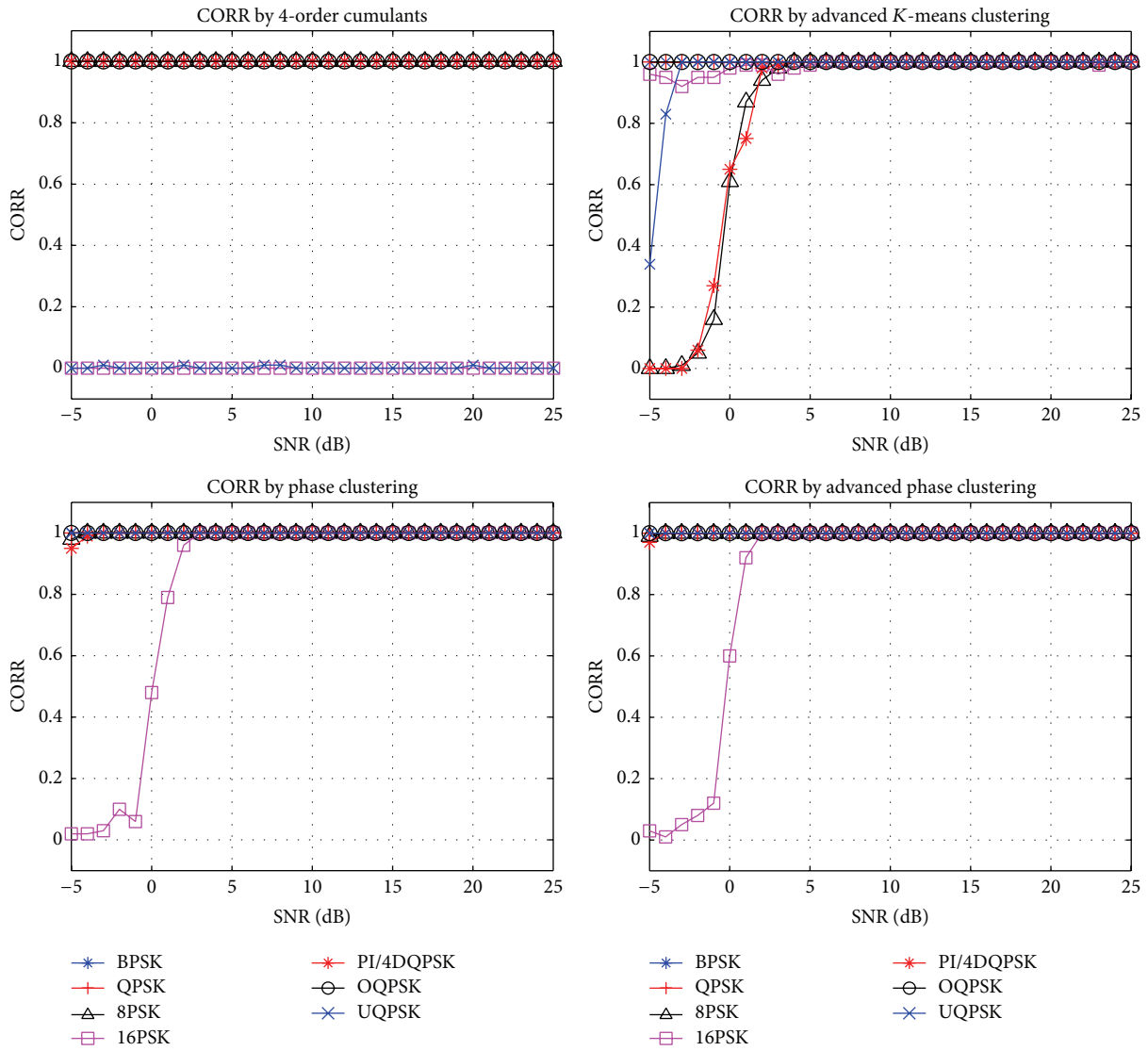


Figure 2: Comparison of correct order recognition rate.

range as WNALP, which has proved to be a better choice for a large estimation range than other algorithms. This estimation range cannot be increased even if the SNR increases. It can be seen that the proposed method in this paper can achieve a large frequency offset estimation range.

Figure 7 vividly shows the difference of constellation changes before and after frequency offset correction by two distinct colors. Signal with frequency offset make its constellation is displayed as a blue ring, which cannot catch any constellation point. However, the approximate original appearance is displayed after frequency offset correction with red. It can be seen that the proposed method in this paper has a remarkable effectiveness under 0 dB.

5. Conclusion

This paper presents a robust classifier for NDA recognition in noncooperative wireless environment. Nonsupervised clustering of the signal phase is achieved by measuring the radian distance between each signal phase and the reference phase. This method proposed optimizes the clustering function and reduces the computation sharply. Moreover, frequency offset is considered and an advanced method is proposed for frequency offset estimation and correction. First, a normalization of the baseband signal is performed. After the nonlinear operation of raising the signal to a power of Q, the estimate of frequency offset is obtained via the weighted summation of the differential phase of the signal's autocorrelation. This method balances estimation accuracy and range under low SNR conditions, which sharply improves the estimation accuracy without shrinking the maximum estimation range, even if the SNR is as low as −15 dB. Seven common PSK signals are adopted for simulation experiments. The classification performance and the estimation and correlation for frequency offset are displayed and demonstrated with several simulation result figures, which illustrate their feasibility and practice.

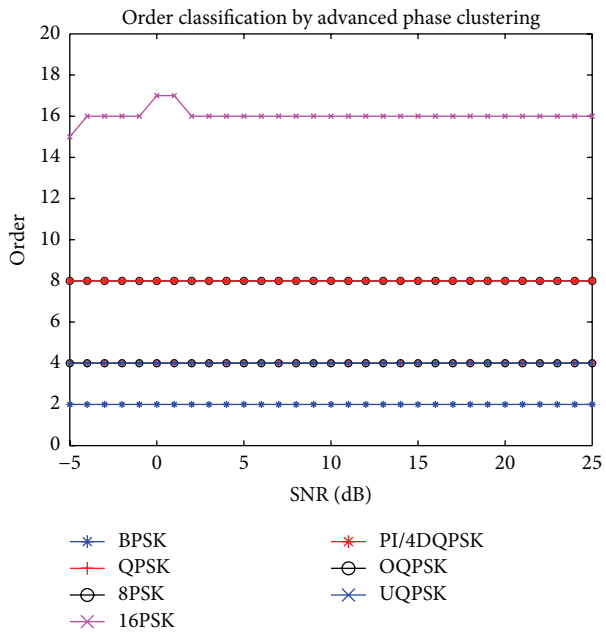


Figure 3: Order classification by advanced phase clustering.

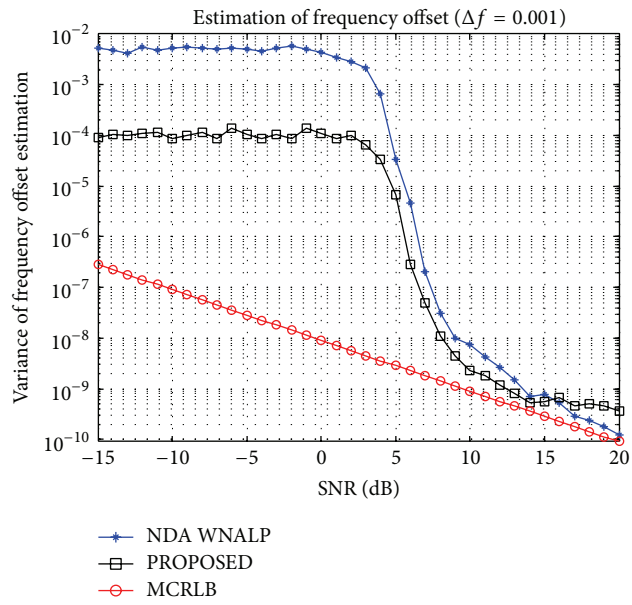


Figure 5: Comparison of estimation variance of frequency offset.

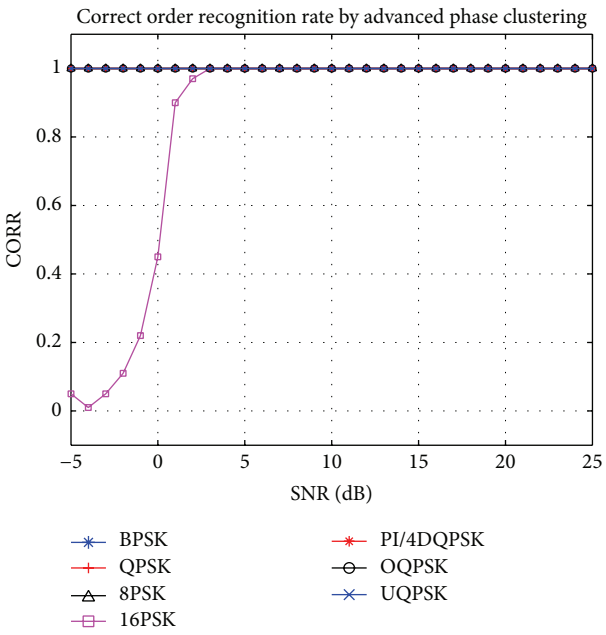


Figure 4: Correct order recognition rate by advanced phase clustering.

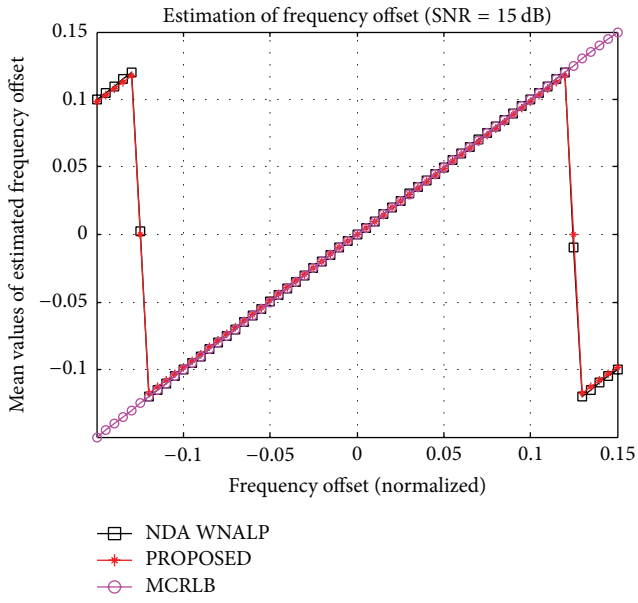


Figure 6: Estimated range comparison.

Generally, this classifier, which is derived from data mining and image processing, has a guiding value for signal processing in electronic surveillance and electronic countermeasure of communication networks. Further work, such as the initial optimization of clustering centers θ and multipath and Rayleigh channel and other practical problems, is considered for applications.

Competing Interests

The authors declare that they have no competing interests.

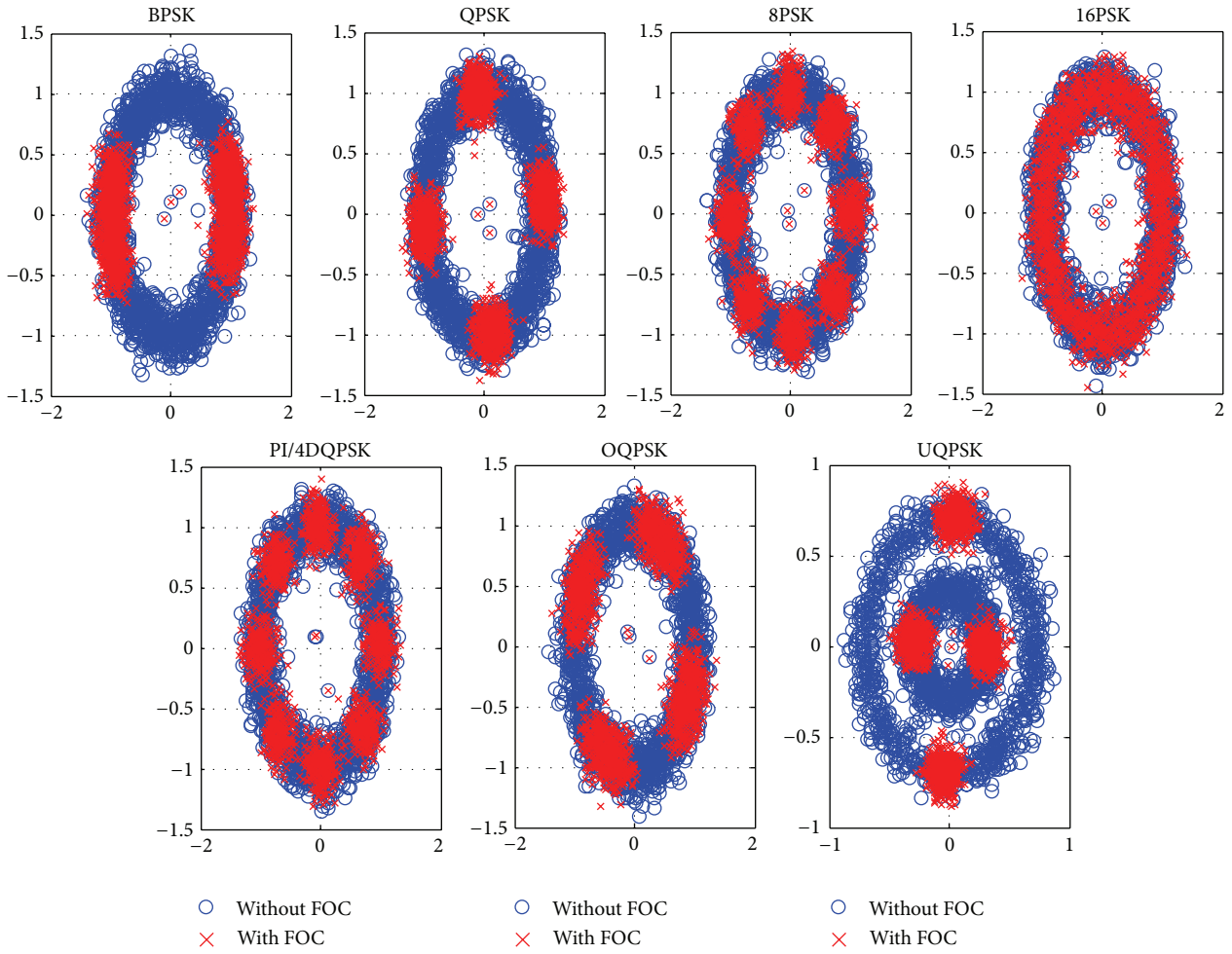


Figure 7: Comparison of frequency offset correction.

Acknowledgments

The authors would like to thank the support of Fundamental Research Funds for the Central Universities (designation: NSFC61371184).

References

[1] F. F. Liedtke, "Computer simulation of an automatic classification procedure for digitally modulated communication signals with unknown parameters," *Signal Processing*, vol. 6, no. 4, pp. 311–323, 1984.

[2] Q. Shi and Y. Karasawa, "Automatic modulation identification based on the probability density function of signal phase," *IEEE Transactions on Communications*, vol. 60, no. 4, pp. 1033–1044, 2012.

[3] A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," *IEEE Transactions on Communications*, vol. 46, no. 4, pp. 431–436, 1998.

[4] K. Kim and A. Polydoros, "Digital modulation classification: the BPSK versus QPSK case," in *Proceedings of the IEEE Military Communications Conference, Conference Record. 21st Century Military Communications—What's Possible? (MILCOM '88)*, pp. 431–436, San Diego, Calif, USA, October 1988.

[5] K. Kim and A. Polydoros, "On the detection and classification of quadrature digital modulations in broad-band noise," *IEEE Transactions on Communications*, vol. 38, no. 8, pp. 1199–1211, 1990.

[6] C. S. Long, K. M. Chugg, and A. Polydoros, "Further results in likelihood classification of QAM signals," in *Proceedings of the IEEE Military Communications Conference (MILCOM '94)*, vol. 1, pp. 57–61, IEEE, Fort Monmouth, NJ, USA, October 1994.

[7] Y.-C. Lin and C.-C. J. Kuo, "Classification of quadrature amplitude modulated (QAM) signals via sequential probability ratio test (SPRT)," *Signal Processing*, vol. 60, no. 3, pp. 263–280, 1997.

[8] W. A. Gardner and C. M. Spooner, "Cyclic spectral analysis for signal detection and modulation recognition," in *Proceedings of the IEEE Military Communications Conference*, pp. 419–424, San Diego, Calif, USA, October 1988.

[9] L. Hong and K. C. Ho, "Identification of digital modulation types using the wavelet transform," in *Proceedings of the Military Communications Conference (MILCOM '99)*, pp. 427–431, Atlantic City, NJ, USA, November 1999.

[10] K. C. Ho, W. Prokopiw, and Y. T. Chan, "Modulation identification of digital signals by the wavelet transform," *IEE Proceedings—Radar, Sonar and Navigation*, vol. 147, no. 4, pp. 169–176, 2000.

[11] C. Cui, H. Li, and J. Yu, "MPSK modulation classification based on morlet transform," *Communication Technology*, vol. 43, no. 3, pp. 10–12, 2010.

[12] V. D. Orlic and M. L. Dukic, "Automatic modulation classification algorithm using higher-order cumulants under real-world channel conditions," *IEEE Communications Letters*, vol. 13, no. 12, pp. 917–919, 2009.

[13] J.-F. Wang, Y. Yue, and J. Yao, "A MPSK recognition method based on high order cumulants," *Communication Technology*, vol. 9, no. 43, pp. 4–6, 2010.

[14] J.-X. Wang and H. Song, "Digital modulation recognition based on constellation diagram," *Journal of China Institute of Communications*, vol. 25, no. 6, pp. 166–173, 2004.

[15] R. Ghunhui, W. Ping, and X. Xianci, "Classification of MPSK signals using the distribution of in-phase and quadrature signals," *Signal Processing*, vol. 22, no. 6, pp. 787–790, 2006.

[16] T.-J. Lu, S.-B. Guo, and X.-C. Xiao, "Fractal characteristics research of modulated signals," *Science in China, Series E: Technological Sciences*, vol. 31, no. 6, pp. 508–510, 2001.

[17] C. Weber, M. Peter, and T. Felhauer, "Automatic modulation classification technique for radio monitoring," *Electronics Letters*, vol. 51, no. 10, pp. 794–796, 2015.

[18] J.-F. Xu, F.-P. Wang, and Z.-J. Wang, "MPSK modulation recognition method based on phase clustering," *Journal of Circuits and Systems*, vol. 16, no. 5, pp. 55–58, 2011.

[19] X.-Y. Chen, Y.-F. Min, L.-R. Zheng, and L. Yang, "Quick mountain clustering algorithm," *Application Research of Computers*, vol. 25, no. 7, pp. 2043–2045, 2008.

[20] N. Ahmadi, "Using fuzzy clustering and TTSAS algorithm for modulation classification based on constellation diagram," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 3, pp. 357–370, 2010.

[21] D. C. Rife and R. R. Boorstyn, "Single-tone parameter estimation from discrete-time observations," *IEEE Transactions on Information Theory*, vol. IT-20, no. 5, pp. 591–598, 1974.

[22] A. B. Awoseyila, C. Kasparis, and B. G. Evans, "Improved single frequency estimation with wide acquisition range," *Electronics Letters*, vol. 44, no. 3, pp. 245–247, 2008.

Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 4356127, 8 pages
http://dx.doi.org/10.1155/2016/4356127

Research Article

A High-Order CFS Algorithm for Clustering Big Data

Fanyu Bu,[1,2] Zhikui Chen,[1] Peng Li,[1] Tong Tang,[3] and Ying Zhang[4]

[1]*School of Software Technology, Dalian University of Technology, Dalian 116620, China*
[2]*School of Computer Information Management, Inner Mongolia University of Finance and Economics, Hohhot 010070, China*
[3]*Department of Student Work, Southwest University, Chongqing 400715, China*
[4]*College of Business Administration, Dalian University of Finance and Economics, Dalian 116622, China*

Correspondence should be addressed to Fanyu Bu; bufanyu@imufe.edu.cn

Received 6 May 2016; Accepted 26 June 2016

Academic Editor: Beniamino Di Martino

With the development of Internet of Everything such as Internet of Things, Internet of People, and Industrial Internet, big data is being generated. Clustering is a widely used technique for big data analytics and mining. However, most of current algorithms are not effective to cluster heterogeneous data which is prevalent in big data. In this paper, we propose a high-order CFS algorithm (HOCFS) to cluster heterogeneous data by combining the CFS clustering algorithm and the dropout deep learning model, whose functionality rests on three pillars: (i) an adaptive dropout deep learning model to learn features from each type of data, (ii) a feature tensor model to capture the correlations of heterogeneous data, and (iii) a tensor distance-based high-order CFS algorithm to cluster heterogeneous data. Furthermore, we verify our proposed algorithm on different datasets, by comparison with other two clustering schemes, that is, HOPCM and CFS. Results confirm the effectiveness of the proposed algorithm in clustering heterogeneous data.

1. Introduction

With the rapid development of the Internet of Things, Internet of People, and Industrial Internet, big data analytics and mining have become a hot topic [1]. One widely used technique of big data analytics and mining is clustering that aims to group data into several clusters according to similarities between the data objects [2]. In 2014, Laio and Rodriguez proposed a novel clustering algorithm by fast search and finding of density peaks (CFS) published in Science Magazine [3]. CFS is the most potential clustering technique because of its efficiency and high accuracy. However, CFS is limited in clustering big data because it cannot cluster heterogeneous data which is prevalent in big data.

Heterogeneous data, different from the homogeneous data containing only one type of objects, involves multiple interrelated types of objects [4]. Moreover, a heterogeneous data object is usually of complex correlations among different modalities. Therefore, heterogeneous data poses important challenges on clustering techniques. Recently, researchers have proposed some algorithms to cluster heterogeneous data [5]. One of this type is based on the graph partition, for instance, the bipartite spectral algorithm, which clusters heterogeneous data by optimizing a unified objective function. However, this kind of methods is usually of low efficiency for clustering big datasets since they need to solve an eigen-decomposition procedure. Another typical algorithm based on the nonnegative matrix factorization, such as SS-NMF, clusters heterogeneous data by revealing the relationships between different objects in a semantic space. In addition, Comrafs is developed for clustering heterogeneous data by constructing the Markov Rand Fields. Since this method is of high computational complexity, it is limited for large-scale heterogeneous data clustering. These algorithms could cluster heterogeneous data; however, they are hard to achieve desired clustering results since they do not model the high nonlinear correlations over multiple types of heterogeneous data objects effectively. Moreover, they are of high time complexity, leading to low efficiency in clustering heterogeneous data.

In this paper, we propose a high-order CFS algorithm (HOCFS) for clustering heterogeneous data based on the dropout deep learning model. The dropout deep learning model was proposed by Hinton to prevent overfitting [6]. It is especially useful in training large networks with small

amount of samples. However, the dropout sets the same omitting probability with 0.5 in each hidden layer of the deep learning model, resulting in its ineffectiveness. Aiming at this problem, we propose an adaptive dropout deep learning model, which sets the omitting probability of each hidden layer according to the relationship between the omitting probability and the layer opposition. Then, we applied the proposed adaptive dropout deep learning model in feature learning for each type of data of every heterogeneous data object. Next, the algorithm uses the vector outer product to fuse the learned features to form a feature tensor for each heterogeneous data object. Finally, since the tensor distance can not only measure the distance between every two heterogeneous samples but also reveal the intrinsic correlations between different coordinates in the high-order tensor space, the tensor distance is applied to the CFS algorithm for clustering heterogeneous data represented by fused features.

Finally, we compare our proposed algorithm with two representative data clustering techniques, namely, HOPCM and CFS, on two datasets, namely, NUS-WIDE and CUAVE in terms of E^* and Rand Index (RI).

Therefore, the contributions of the paper are summarized as the following three aspects:

(i) Current dropout deep learning models are of low effectiveness and efficiency in learning features for heterogeneous data. To tackle this problem, the paper proposes an adaptive dropout deep learning model to learn features for each type of data and then fuses the learned features to form a feature tensor for each heterogeneous data object.

(ii) To measure the similarity between heterogeneous data objects in high-order tensor space, the paper applies the tensor distance in the clustering process.

(iii) Conventional CFS algorithm cannot cluster heterogeneous data directly because it works in the vector space. The paper extends the CFS algorithm from the vector space to the tensor space for clustering heterogeneous data represented by the feature tensors.

2. Preliminaries

This section presents the technique preliminaries about our scheme, including the stacked autoencoder, dropout, and the CFS clustering algorithm. The stacked autoencoder is presented first, followed by the CFS clustering algorithm.

2.1. Stacked Autoencoder (SAE) and Dropout. The stacked autoencoder (SAE) that is one important example of deep learning models has been widely employed in supervised feature learning for many applications [7]. SAE is built to learn hierarchical features of data by stacking multiple basic autoencoders (BAEs) as shown in Figure 1.

As the typical module of a stacked autoencoder, a basic autoencoder (BAE) [8] learns a hidden representation h of the input data x by an encoding function f:

$$h = f_\theta\left(W^{(1)}x + b^{(1)}\right). \tag{1}$$

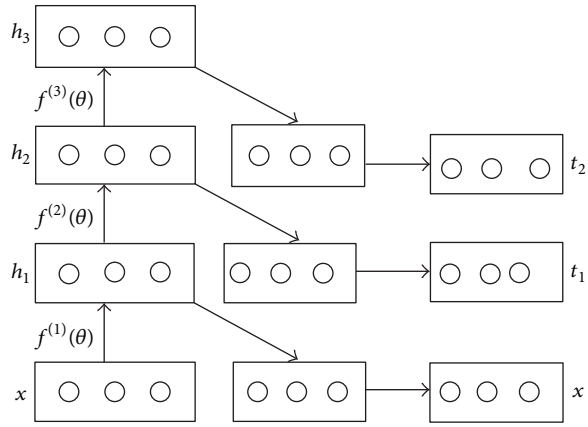


Figure 1: The architecture of the stacked autoencoder.

Then, BAE reconstructs the input from the hidden representation h to a reconstruction y by a decoding function s:

$$y = s_\theta\left(W^{(2)}x + b^{(2)}\right), \tag{2}$$

where $\theta = (W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)})$ denotes the parameter of the autoencoder and the functions f and s typically adopt the sigmoid function: $f(x) = 1/(1 + e^{-x})$.

To train the parameter of the autoencoder, an objective function with a weight-decay that is used to prevent overfitting is defined as follows:

$$J(\theta) = \left(\sum_{x\in D} L\left(x, s\left(f\left(x\right)\right)\right)\right) + \lambda\sum_{ij} W_{ij}^2, \tag{3}$$

where L is the reconstruction error and λ is a hyperparameter used to control the strength of the regularization.

The stacked autoencoder is a full-connected model and it involves many redundant connections. Therefore, it usually produces overfitting in the real applications. Aiming at this problem, Hinton proposed dropout to reduce the overfitting by preventing coadaption of feature detectors in deep learning models. It randomly omits half of the feature detectors on each training sample to prevent a hidden unit from relying on other hidden units being present. Dropout was proved to be especially effective and efficient in training a large neural network with a small training set.

2.2. Clustering by Fast Search and Finding of Density Peaks (CFS). CFS is the latest clustering algorithm proposed by Laio and Rodriguez in Science Magazine in 2014 [3]. It is highly robust and efficient. More importantly, it can find clusters of arbitrary shape and determine the number of clusters automatically. Several experiments have demonstrated its superiority in the efficiency and effectiveness over the previous algorithms for clustering large amounts of data. Therefore, it has become the most potential algorithm for clustering big data.

The key of the CFS algorithm lies in the characterization of cluster centers. Particularly, the algorithm basically assumes that cluster centers should be surrounded by neighbor objects with lower local density and be more far away

from other objects with a higher local density. Based on this assumption, CFS defines two quantities for every data object x_i, the local density ρ_i and the minimum distance δ_i from any other object with higher density, in

$$\rho_i = \sum_j \chi\left(d_{ij} - d_c\right)$$

$$\chi(x) = \begin{cases} 1 & \text{if } x < 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$\delta_i = \min_{j:\rho_j > \rho_i}\left(d_{ij}\right),$$

where d_c represents a cutoff distance. According to Laio and Rodriguez, d_c can be set to the biggest 2% of all the distances between every two objects to get a good clustering result. For the object x_i with the highest density, its distance δ_i is taken as $\delta_i = \max_j(d_{ij})$.

In the CFS algorithm, cluster centers are recognized as the objects with the large value of γ that is defined in

$$\gamma_i = \rho_i \times \delta_i. \tag{5}$$

3. Problem Statement

Consider a dataset with n heterogeneous data objects $X = \{x_1, x_2, \ldots, x_n\}$ and assume that each object can be represented by a feature tensor. The task of heterogeneous data clustering is to classify the dataset into groups according to their similarity such that the objects belonging to the same cluster share similarity as much as possible. Based on the analysis in the previous parts, heterogeneous data poses a large number of challenges on the clustering techniques. We discuss the key issues in three following aspects:

(1) *Feature Learning of Heterogeneous Data*. Feature learning is the fundamental step for heterogeneous data clustering. In fact, many feature learning algorithms, especially some methods based on deep learning, have been well studied in recent years. However, most of them are hard to learn features for heterogeneous data. Although the deep computation model can learn features for heterogeneous data, it is of low accuracy and efficiency since it cannot avoid overfitting.

(2) *Similarity Measurement for Heterogeneous Data*. Similarity measurement is the key to one clustering technique. There are a lot of metrics for measuring the similarity between two objects. However, they can only measure the distance between homogeneous objects represented by feature vectors because they work in the vector space. A heterogeneous object is typically represented by a feature tensor, making most of current metrics hard to calculate the similarity for heterogeneous data objects.

(3) *Clustering Technique for Heterogeneous Data*. Typically, a heterogeneous object is represented by a feature tensor. However, most of clustering techniques including the CFS algorithm work only in the vector space, resulting in failure to cluster heterogeneous data in the high-order tensor space.

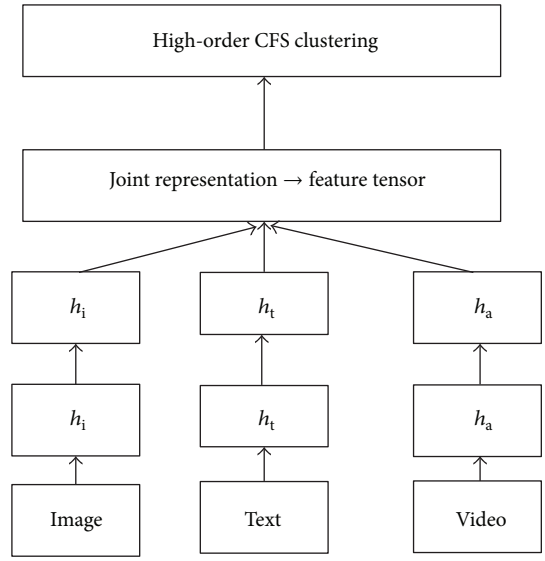


Figure 2: The architecture of the proposed scheme.

4. High-Order CFS Algorithm Based on Dropout Deep Computation Model

In this section, we describe the details of the proposed high-order CFS algorithm based on the dropout deep learning model for clustering heterogeneous data. The proposed algorithm works in three stages: unsupervised feature learning, feature fusion, and high-order clustering, which is shown in Figure 2.

In the first stage, each type of data in the heterogeneous dataset is separately learned by the proposed adaptive dropout deep learning model. In the second stage, the proposed algorithm uses the vector outer product to fuse the learned futures to form a feature tensor as the joint representation of each object. Finally, the proposed algorithm extends the conventional CFS technique from the vector space to the tensor space for clustering the heterogeneous dataset.

4.1. Feature Learning Based on the Adaptive Dropout Deep Learning Model. In the dropout deep learning model, each hidden unit is randomly omitted from the network always with a constant probability of 0.5. This way will ignore the relationship between the omitting probability and the layer opposition, resulting in a low effectiveness of deep learning models in heterogeneous data feature learning. A large number of studies demonstrate that the fundamental layers of a deep architecture share many common characters, implying that the dropout in the lower layers has more generalization function than that in higher layers. Therefore, the omitting probability of the dropout should decay with the layers becoming higher.

Based on the above analysis, we propose an adaptive dropout deep learning model by defining a distribution model of the omitting probability y of dropout as the following function:

$$y = f(l) = \begin{cases} -0.1l + 0.05n + 0.5 & n = 2k \ (k = 1, 2, \ldots) \\ -0.1l + 0.05n + 0.55 & n = 2k - 1 \ (k = 1, 2, \ldots), \end{cases} \quad (6)$$

where $n \leq 9$ denotes the number of hidden layers in the deep learning model and l represents the position of the layer.

Function (6) has the following properties:

(1) it is monotonically decreasing.

(2) The omitting probability is 0.5 for the middle hidden layer.

(3) The omitting probability is always in $(0, 1)$ for $x = 1, 2, \ldots, n$.

Proof. (1) By the assumption, function $f(l)$ is continuously differentiable and we may write

$$f'(l) = -0.1 < 0, \quad (7)$$

which implies that (6) is a strictly decreasing function. Particularly, the omitting probability of the dropout should decay with the layers becoming higher.

(2) When $n = 2k \ (k = 1, 2, \ldots)$,

$$f\left(\frac{n}{2}\right) = -0.1 \times \frac{n}{2} + 0.05 \times n + 0.5 = 0.5. \quad (8)$$

When $n = 2k - 1 \ (k = 1, 2, \ldots)$,

$$f\left(\frac{n+1}{2}\right) = -0.1 \times \frac{n+1}{2} + 0.05 \times n + 0.55 = 0.5, \quad (9)$$

which proves that the omitting probability is 0.5 for the middle hidden layer.

(3) Based on property (1),

$$f(n) \leq f(l) \leq f(1) \quad (1 \leq n \leq 9). \quad (10)$$

Then,

$$f(n) = \begin{cases} -0.05n + 0.5 \geq -0.05 \times 8 + 0.5 = 0.1 > 0 \\ -0.05n + 0.55 \geq -0.05 \times 9 + 0.55 = 0.1 > 0 \end{cases} \quad (11)$$

$$f(1) = 0.05n + 0.45 \leq 0.05 \times 9 + 0.45 = 0.9 < 1.$$

Therefore, the omitting probability is always in $(0, 1)$ for $l = 1, 2, \ldots, n$. □

We can get the adaptive dropout deep learning model by applying the distribution function of the omitting probability to the deep learning model outlined in Algorithm 1.

In the proposed high-order CFS algorithm, the adaptive dropout deep learning model is used to learn features of each type of data of the heterogeneous data.

Input: $\{(X^{(i)}, Y^{(i)})\}, 1 \leq i \leq N$, iterater$_{\max}$, η, threshold
Output: $\theta = \{W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)}\}$

(1) Randomly initialize all $\theta = \{W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)}\}$;
(2) $y = f(l)$;
(3) **for** iteration $= 1, 2, \ldots,$ iterater$_{\max}$ **do**
(4) **for** example $= 1, 2, \ldots, N$ **do**
(5) **for** $j = 1, 2, \ldots, m$ **do**
(6) $z_j^{(2)} = w_{ji}^{(1)} \cdot x_i + b_j^{(1)}$;
(7) $a_j^{(2)} = f(z_j^{(2)})$;
(8) mark$\{i\}$ = rand(size$(a^{(2)}) > y$);
(9) $a^{(2)} = a^{(2)} \cdot *$mark$\{i\}$;
(10) **for** $i = 1, 2, \ldots, n$ **do**
(11) $z_i^{(3)} = w_{ij}^{(2)} \cdot a_j^{(2)} + b_i^{(2)}$;
(12) $a_i^{(3)} = f(z_i^{(3)})$;
(13) **for** $i = 1, 2, \ldots, n$ **do**
(14) $\sigma_i^{(3)} = -(y - a_i^{(3)}) \cdot f'(z_i^{(3)})$;
(15) **for** $j = 1, 2, \ldots, m$ **do**
(16) $\sigma_j^{(2)} = (\sum_{i=1}^{n} w_{ij}^{(2)} \cdot \sigma_i^{(3)}) \cdot f'(z_j^{(2)})$;
(17) $\sigma^{(2)} = \sigma^{(2)} \cdot *$[ones(size$(\sigma^{(2)}, 1), 1)$mark$\{i\}$];
(18) **for** $i = 1, 2, \ldots, n$ **do**
(19) $b_i^{(2)} = \sigma_i^{(3)}$;
(20) **for** $j = 1, 2, \ldots, m$ **do**
(21) $\Delta w_{ij}^{(2)} = a_j^{(2)} \cdot \sigma_i^{(3)}$;
(22) **for** $j = 1, 2, \ldots, m$ **do**
(23) $b_j^{(1)} = \sigma_j^{(2)}$;
(24) **for** $i = 1, 2, \ldots, n$ **do**
(25) $\Delta w_{ji}^{(1)} = x_i \cdot \sigma_j^{(2)}$;

Algorithm 1: Adaptive dropout backpropagation neural network learning algorithm.

4.2. Feature Fusion Using Vector Outer Product. The vector outer product is one of the widely used operations in mathematics, denoted by ⊗. If A is an m-dimension vector and B is an n-dimension vector, their outer product will produce an $m \times n$ matrix C; $C = A \otimes B$. Each entry in the matrix C is defined as $c_{ij} = a_i \cdot b_j$, where a_i and b_j are one entry in vectors A and B, respectively. One example of the vector outer product is as shown in (6):

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \otimes \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} = \begin{bmatrix} a_1b_1 & a_2b_1 & a_3b_1 \\ a_1b_2 & a_2b_2 & a_3b_2 \\ a_1b_3 & a_2b_3 & a_3b_3 \\ a_1b_4 & a_2b_4 & a_3b_4 \end{bmatrix}. \quad (12)$$

More generally, the outer product of n vectors $A_1 \in R^{I_1}$, $A_2 \in R^{I_2}, \ldots, A_n \in R^{I_n}$ will produce an n-order tensor $B \in R^{I_1 \times I_2 \times \cdots \times I_n}$, $B = A_1 \otimes A_2 \otimes \cdots \otimes A_n$, in which each entry is defined as $b_{i_1 i_2 \cdots i_n} = a_{1_{i1}} \cdot a_{2_{i2}} \cdot \ldots \cdot a_{n_{in}}$.

After using the adaptive deep learning model to learn features of heterogeneous data, each type of data can be represented by a feature vector. Particularly, for the heterogeneous dataset in which each object consists of one image, one text, and one piece of video, three vectors, a, b, and c, are used to represent the feature vectors learned from the adaptive

Input $X = \{X_1, X_2, \ldots, X_N\}, d_c$
Output cl[n], center[k]
(1) **for** $i = 1, 2, \ldots, n$ **do**
(2) **for** $j = i + 1, i + 2, \ldots, n$ **do**
(3) $d_{ij} = \sqrt{(X_i - X_j)^T G (X_i - X_j)}$;
(4) **for** $i = 1, 2, \ldots, n$ **do**
(5) $\rho_i = \sum_j \chi(d_{ij} - d_c)$;
(6) **for** $i = 1, 2, \ldots, n$ **do**
(7) $\delta_i = \min_{j:\rho_j > \rho_i} \{d_{ij}\}$;
(8) $\gamma_i = \rho_i \times \delta_i$;
(9) Select clustering centers according to γ_i;
(10) **for** $i = 1, 2, \ldots, n$ **do**
(11) cl[i] = $\min_{j:\text{centers}[k]} \{d_{ij}\}$;

Algorithm 2: High-order CFS clustering algorithm.

dropout deep learning model, respectively. In this subsection, such feature vectors are fused by the vector outer product to form one feature tensor X for joint representation of one object in the heterogeneous dataset according to the following rules:

(1) For the object with only one image and one text, its feature tensor is represented by $X = a \otimes b$.

(2) For the object with only one image and one piece of video, its feature tensor is represented by $X = a \otimes c$.

(3) For the object with only one text and one piece of video, its feature tensor is represented by $X = b \otimes c$.

(4) For the object with only one image, one text, and one piece of video, its feature tensor is represented by $X = a \otimes b \otimes c$.

4.3. The High-Order CFS Clustering. As discussed in Section 2, the conventional CFS algorithm cannot cluster heterogeneous data directly because it works in the vector space while each object in the heterogeneous dataset is represented by a feature tensor. To tackle this problem, we propose a high-order CFS algorithm for clustering heterogeneous data.

To calculate the distance between two points in high-order tensor space, represented by two tensors, $X, Y \in R^{I_1 \times I_2 \times \cdots \times I_N}$, they need to be unfolded to the corresponding vectors. In detail, the item $X_{i_1 i_2 \cdots i_N}$ is unfolded to x_l by $l = i_1 + \sum_{j=2}^{N} \prod_{t=1}^{j-1} I_t$.

The proposed high-order CFS clustering algorithm (HOCFS) based on the feature tensor is outlined in Algorithm 2.

5. Performance Evaluation of Adaptive Dropout Model

In this part, we assess the adaptive dropout deep learning model on the STL-10 and CIFAR-10 datasets by comparison with the conventional dropout model.

5.1. Experiments on the STL-10 Dataset. We initially explored the effectiveness of adaptive dropout using STL-10, a widely used benchmark for machine learning algorithms. It contains 500 training images, 800 testing images that are grouped by 10 classes, and 100000 unlabeled images for unsupervised learning. We combine the adaptive dropout distribution model with stacked autoencoders to train two deep learning models. One has 4 hidden layers while the other has 5 hidden layers. Both of them have one logistic regression layer on the top. For the adaptive dropout deep learning model, we use the proposed algorithm to set the omitted rate of hidden units while setting omitted rate of 0.5 of hidden units for the conventional dropout deep learning model. The classification results are presented in Figures 3 and 4.

From Figures 3 and 4, the classification error decreases with the epoch increasing. The classification error produced by the adaptive dropout model is lower than that produced by the conventional dropout model. Particularly, we achieved

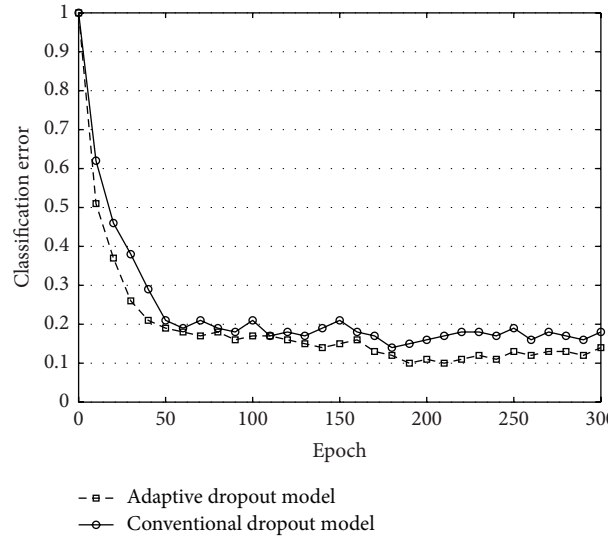


Figure 3: Classification result on STL-10 with 4 hidden layers.

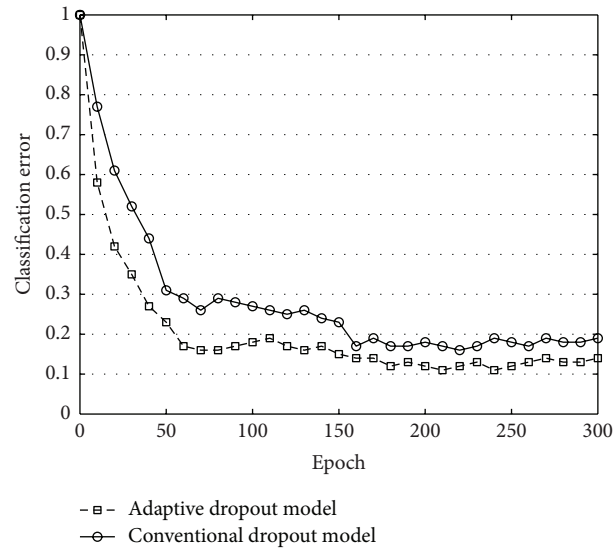


Figure 4: Classification result on STL-10 with 5 hidden layers.

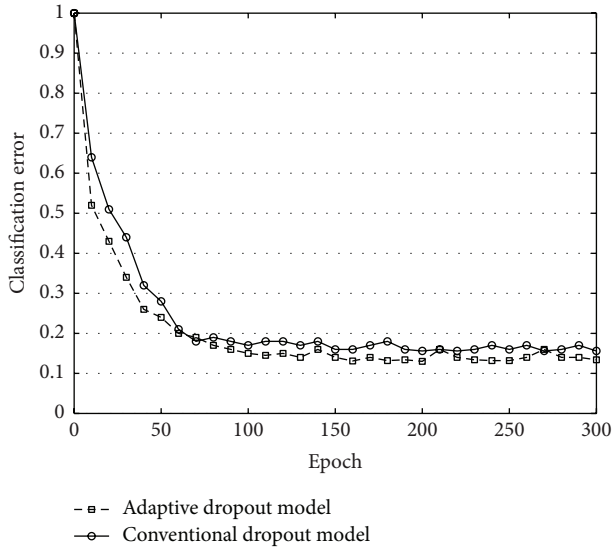


Figure 5: Classification result on CIFAR-10.

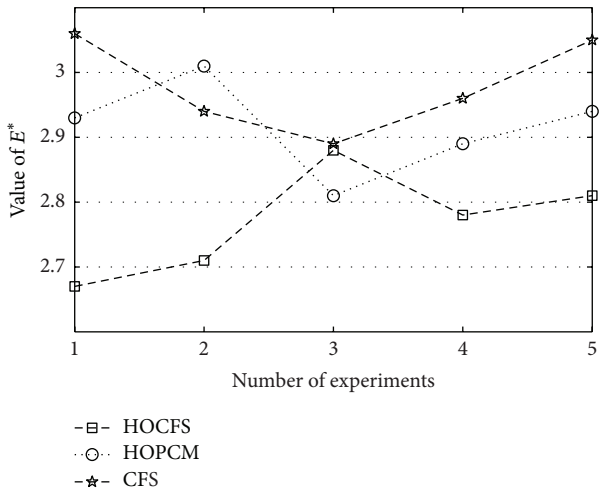


Figure 6: Clustering result on NUS-WIDE in terms of E^*.

the best classification error rate of 0.10 by using adaptive dropout model with 4 hidden layers while the best classification error rate given by the conventional dropout model is 0.12, which indicates that our proposed model performs better than the conventional dropout model in classifying the STL-10 dataset.

5.2. Experiments on the CIFAR-10 Dataset. CIFAR-10 is a benchmark task for object recognition, consisting of 60000 color images in 10 groups, with 6000 images per group. These images were labeled by hand to produce 50000 training images and 10000 test images. We built a classification network with three convolutional layers and three pooling and two fully connected layers to explore the effectiveness of the adaptive dropout model on CIFAR-10 dataset. Each convolutional layer has an exclusive ReLU layer and a dropout layer. Specially for the adaptive dropout deep learning model, we use the proposed algorithm to set the omitted rate of hidden units while setting omitted rate of 0.5 of hidden units for the conventional dropout deep learning model. The classification results are presented in Figure 5.

From Figure 5, the error rate produced by the adaptive dropout model is lower than that produced by the conventional dropout model in most cases. More importantly, using the conventional dropout model gives the best error rate of 0.156. This is reduced to 0.136 by using the adaptive dropout model, which implies that the proposed model works much better than the conventional dropout model for CIFAR-10.

6. Performance Evaluation of the High-Order CFS Algorithm

In this part, we evaluate the high-order CFS clustering algorithm by comparison with the HOPCM algorithm and the conventional CFS algorithm on two representative heterogeneous datasets, namely, NUS-WIDE and CUAVE, in terms of E^* and Rand Index (RI).

HOCPM was developed in 2015 for clustering heterogeneous data by combining the autoencoder model and the possibilistic c-means algorithm [9]. For the conventional CFS algorithm, we perform the same preprocessing step with our proposed algorithm. Particularly, we first use the adaptive dropout deep learning model to learn features of texts, audios, and images of each object and then form a feature vector for the object by concatenating the learned features. Finally, the Euclidean distance is applied for the conventional CFS algorithm to cluster the heterogeneous dataset where each object is represented by a learned feature vector.

The evaluation criteria are described in Section 6.1, followed by the experimental results.

6.1. Experiments on the NUS-WIDE Dataset. The NUS-WIDE dataset is the biggest image set, consisting of 269, 648 annotated images. To compare the proposed algorithm with the HOPCM algorithm and the conventional CFS algorithm fairly, we use the same image dataset collected from the NUS-WIDE with literature [9], which consists of 8 different subsets, each with 10,000 annotated images falling into 14 categories.

First, we carried out the experiments on the overall image set for five times. The clustering results are shown in Figures 6 and 7.

Figure 6 shows the clustering result in terms of E^* on the overall dataset. We observe that the proposed algorithm got the lowest values of E^* in most cases, which implies that the proposed algorithm produced the most accurate clustering centers.

From Figure 7, HOCFS produced the highest values of RI in most cases, which indicates that HOCFS performs best in clustering NUS-WIDE dataset. Moreover, the conventional CFS algorithm performs worst in terms of E^* and RI, demonstrating that the proposed algorithm could effectively capture the complex correlations over the heterogeneous data by applying the vector outer product to feature fusion and using tensor distance to measure the similarity between two objects.

Next, we carried out the experiment on the 8 subsets for 5 times to evaluate the robustness of the clustering algorithms.

Table 1: Clustering result on NUS-WIDE in terms of E^*.

Algorithm/subset	1	2	3	4	5	6	7	8
CFS	2.64	3.01	2.99	3.04	2.73	3.02	3.08	2.82
HOPCM	2.04	2.57	2.91	2.63	2.12	2.91	2.99	2.08
HOCFS	1.96	2.24	2.37	2.28	1.95	2.16	2.39	2.01

Table 2: Clustering result on NUS-WIDE in terms of RI.

Algorithm/subset	1	2	3	4	5	6	7	8
CFS	0.86	0.79	0.87	0.82	0.76	0.79	0.83	0.69
HOPCM	0.91	0.84	0.93	0.91	0.88	0.92	0.82	0.84
HOCFS	0.95	0.84	0.94	0.95	0.93	0.96	0.89	0.91

Tables 1 and 2 present the average clustering results of 5 times on every subset.

From Tables 1 and 2, the average values of E^* obtained by HOCFS are lowest for each subset while the average values of RI obtained by HOCFS are significantly larger than that obtained by HOPCM and CFS. In other words, the proposed algorithm produced the best clustering results in terms of E^* and RI for NUS-WIDE dataset.

6.2. Experiments on the CUAVE Dataset. CUAVE is a typical multimodal dataset consisting of some digits, 0 to 9, reported by 36 individuals. To assess HOCFS for clustering heterogeneous data, we added some annotations to each object as the literature [9].

We first carried out the experiment on the CUAVE dataset for 5 times to judge HOCFS for clustering heterogeneous data in terms of RI. The result is presented in Figure 8.

According to Figure 8, the value of RI obtained by HOCFS is highest for each experiment, implying that the proposed algorithm produced the best clustering result for the CUAVE dataset in terms of RI. On the one hand, the proposed algorithm uses the hybrid stacked autoencoder model to learn features of each object in the CUAVE dataset while HOPCM only uses the basic autoencoder model to learn features, leading to the more accurate clustering result produced by the proposed algorithm compared to HOPCM. On the other hand, HOCFS fuses the learnt features of each modality for capturing the nonlinear correlations over multiple modalities of each object while CFS formed the feature vector for each object by only concatenating the learned features. Thus, the proposed algorithm performed the best for clustering the CUAVE dataset.

Next, we evaluate the robustness of the proposed algorithm by generating three different subsets, each with a distinct combination of two modalities. We carried out the experiment on these subsets for 5 times. The results are shown in Figures 9–11.

According to Figures 9–11, the proposed algorithm outperformed HOPCM and CFS since HOCFS got higher values of RI than the other two algorithms in most cases, especially for clustering the text-audio subset. In other words, the proposed algorithm produced the best clustering results in terms of RI for the CUAVE subsets.

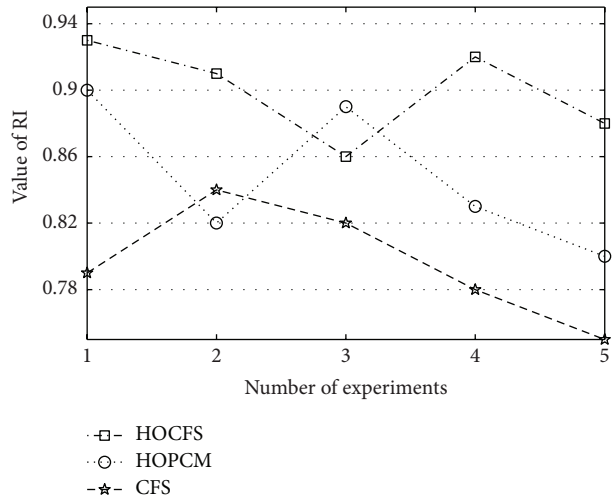


Figure 7: Clustering result on NUS-WIDE in terms of RI.

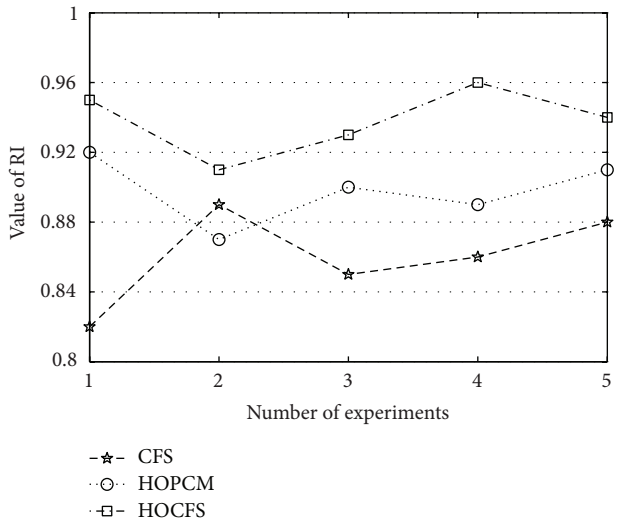


Figure 8: Clustering result on CUAVE in terms of RI.

Finally, we studied the relationship between the clustering result and the different combinations of modalities by analyzing the clustering results, as shown in Table 3.

From Table 3, the best clustering result is always produced on the overall dataset, implying that the clustering result of heterogeneous data relies on the joint features of image-text-audio modalities. Moreover, the proposed algorithm produced the worst clustering result on text-audio subset, which demonstrates that only features learned from the text-audio modalities could not effectively represent the objects in the CUAVE dataset.

7. Conclusion

In this paper, we proposed a high-order CFS algorithm for clustering heterogeneous data. One property of the paper is to devise an adaptive deep learning model and to apply it to learning features of each type of data. Furthermore, the vector

Table 3: Clustering result on different subsets in terms of RI.

Algorithm/subset	1	2	3	4	5
Image-text	0.92	0.88	0.87	0.89	0.93
Text-audio	0.81	0.79	0.78	0.83	0.80
Image-audio	0.89	0.83	0.89	0.85	0.87
Overall	0.96	0.91	0.93	0.96	0.94

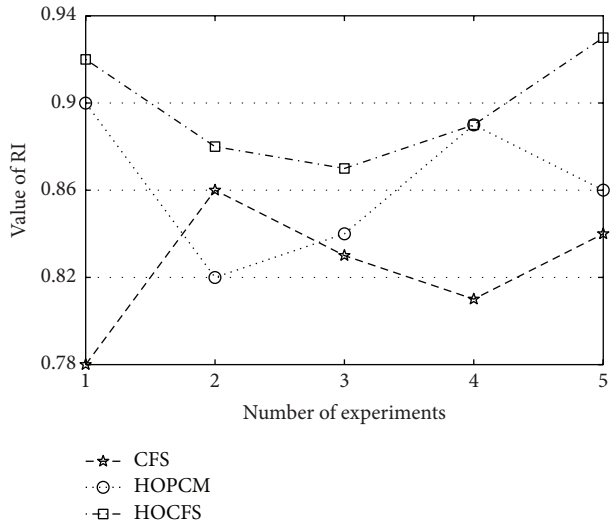


Figure 9: Clustering result on image-text subset in terms of RI.

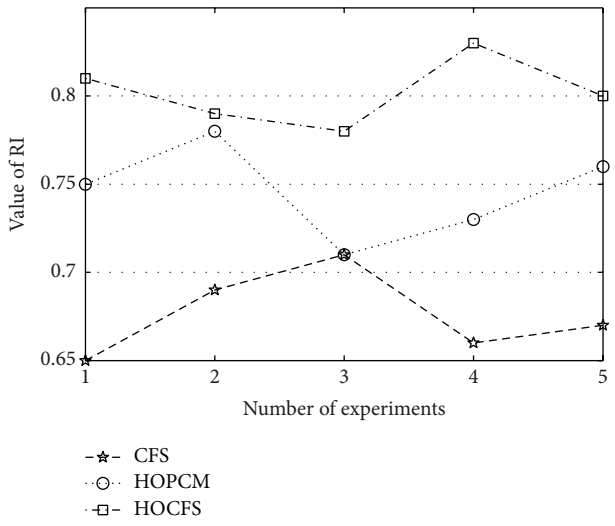


Figure 10: Clustering result on text-audio subset in terms of RI.

outer product was used to model the correlations of each type of data to form a feature tensor for every heterogeneous data object. Another property of the proposed algorithm is to adopt the tensor distance to measure the similarity between every two heterogeneous objects. Experimental results showed that our proposed algorithm produced more accurate results than HOPCM and CFS in terms of E^* and RI.

Recently, more and more complex heterogeneous data have been generated in many applications. For example, there

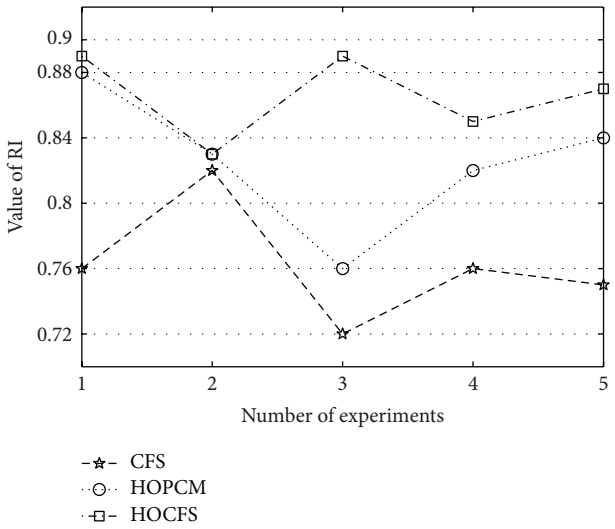


Figure 11: Clustering result on image-audio subset in terms of RI.

are simultaneously many images and audio pieces in one web document. The future work will focus on how to cluster such complex heterogeneous dataset.

Competing Interests

The authors declare that they have no competing interests.

References

[1] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 97–107, 2014.

[2] Q. Zhang and Z. Chen, "A weighted kernel possibilistic c-means algorithm based on cloud computing for clustering big data," *International Journal of Communication Systems*, vol. 27, no. 9, pp. 1378–1391, 2014.

[3] A. Laio and A. Rodriguez, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[4] X. Yang, T. Zhang, and C. Xu, "Cross-domain feature learning in multimedia," *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 64–78, 2015.

[5] L. Meng, A.-H. Tan, and D. Xu, "Semi-supervised heterogeneous fusion for multimedia data co-clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2293–2306, 2014.

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

[7] Q. Zhang, L. T. Yang, and Z. Chen, "Deep computation model for unsupervised feature learning on big data," *IEEE Transactions on Services Computing*, vol. 9, no. 1, pp. 161–171, 2016.

[8] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[9] Q. Zhang, L. T. Yang, Z. Chen, and F. Xia, "A high-order possibilistic-means algorithm for clustering incomplete multimedia data," *IEEE Systems Journal*, 2015.