Wireless Communications and Mobile Computing

Smart Cities: Recent Trends, Methodologies, and Applications

Lead Guest Editor: Damianos Gavalas
Guest Editors: Petros Nicopolitidis, Achilles Kameas, Christos Goumopoulos, Paolo Bellavista, Lampros Lambrinos, and Bin Guo

Smart Cities: Recent Trends, Methodologies, and Applications

Wireless Communications and Mobile Computing

Smart Cities: Recent Trends, Methodologies, and Applications

Lead Guest Editor: Damianos Gavalas
Guest Editors: Petros Nicopolitidis, Achilles Kameas, Christos Goumopoulos, Paolo Bellavista, Lampros Lambrinos, and Bin Guo

Editorial Board

Contents

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 7090963, 2 pages
https://doi.org/10.1155/2017/7090963

Editorial

Smart Cities: Recent Trends, Methodologies, and Applications

Damianos Gavalas,[1] Petros Nicopolitidis,[2] Achilles Kameas,[3] Christos Goumopoulos,[4] Paolo Bellavista,[5] Lampros Lambrinos,[6] and Bin Guo[7]

[1]*Department of Product and Systems Design Engineering, University of the Aegean, Syros, Greece*
[2]*Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece*
[3]*School of Science and Technology, Hellenic Open University, Patras, Greece*
[4]*Department of Information & Communication Systems Engineering, University of the Aegean, Samos, Greece*
[5]*Department of Computer Science and Engineering, University of Bologna, Bologna, Italy*
[6]*Department of Communication and Internet Studies, Cyprus University of Technology, Limassol, Cyprus*
[7]*School of Computer Science, Northwestern Polytechnical University, Xi'an, Shaanxi, China*

Correspondence should be addressed to Damianos Gavalas; dgavalas@aegean.gr

Received 24 September 2017; Accepted 25 September 2017; Published 25 October 2017

Worldwide forecasts indicate that the size and population of cities will increase further. This immense growth will put a strain on resources and pose a major challenge in many aspects of everyday life in urban areas, such as the quality of services in the medical, educational, environmental, transportation, public safety, and security sectors, indicatively. Thus, novel methods of management must be put in place for these cities to remain sustainable. The wide adoption of pervasive and mobile computing systems gave rise to the term of "smart cities," which implies the ability of sustainable city growth by leading to major improvements in city management and life in the above-mentioned sectors and other aspects such as energy efficiency, traffic congestion, pollution reduction, parking space, and recreation. This has been made possible in recent years due to the widespread availability of commodity low-power sensors, smart phones, tablets, and the necessary wireless networking infrastructure, which, along with technologies such as AI and management of big data, may be utilized to address the challenges of sustainable urban environments.

The motivation behind this special issue has been to solicit cutting-edge research relevant to technologies, methodologies, and applications for smart cities. The special issue has attracted 19 submissions. Following a rigorous review process (including a second review round), 7 outstanding papers (acceptance rate 36.8%) have been finally selected for inclusion in the special issue. The accepted papers cover a wide range of research subjects in the broader area of smart cities, including service delivery, service recommendation, user privacy, crowdsensing, and vehicular networks.

The paper "Crowdsensing Task Assignment Based on Particle Swarm Optimization in Cognitive Radio Networks" by L. Zhai and H. Wang proposes an optimal algorithm based on particle swarm optimization to solve the problem of assigning wireless spectrum sensing tasks to mobile intelligent terminals in Cognitive Radio Networks. The algorithm employs crowdsensing principles and takes into account several factors including remaining energy, locations, and costs of mobile terminals.

The paper "An ARM-Compliant Architecture for User Privacy in Smart Cities: SMARTIE—Quality by Design in the IoT" by V. Beltran et al. introduces the IoT-Architecture Reference Model (IoT-ARM) and describes its application within the European-funded project, SMARTIE. The paper discusses the architectural aspects of SMARTIE which support efficient and scalable security and user-centric privacy.

The paper "Fault Activity Aware Service Delivery in Wireless Sensor Networks for Smart Cities" by X. Zhang et al. considers the problem of fault-aware multiservice delivery in Wireless Sensor Network environments, wherein the network performs secure routing and rate control in terms of fault activity dynamic metric. The authors propose a distributed

framework to estimate the fault activity information based on the effects of nondeterministic faulty behaviours and then present a fault activity geographic opportunistic routing (FAGOR) algorithm addressing a wide range of misbehaviours.

The paper "A Hybrid Service Recommendation Prototype Adapted for the UCWW: A Smart-City Orientation" by H. Zhang et al. deals with the problems of cold start and sparsity when considering service recommendation in ubiquitous computing environments. To alleviate these problems, the authors propose a hybrid service recommendation prototype utilizing user and item side information for use in the Ubiquitous Consumer Wireless World (i.e., a novel wireless communication environment that offers a consumer-centric and network-independent service operation model, allowing the materialization of a broad range of smart city scenarios).

The paper "Data Dissemination Based on Fuzzy Logic and Network Coding in Vehicular Networks" by X. Tang et al. presents a data dissemination scheme for vehicular networks based on fuzzy logic and network coding. The scheme addresses the problems of high velocity, frequent topology changes, and limited bandwidth, so as to efficiently propagate data in vehicular networks. Fuzzy logic is used to compute the transmission ability for each vehicle while network coding is utilized to reduce transmission overhead and accelerate data retransmission.

The paper "Unchained Cellular Obfuscation Areas for Location Privacy in Continuous Location-Based Service Queries" by J.-N. Luo and M.-H. Yang describes an unchained regional privacy protection method that combines query logs and chained cellular obfuscation areas to ensure location privacy and effectiveness in location-based services (LBS). The proposed method adopts a multiuser anonymizer architecture to prevent attackers from predicting user travel routes by using background information derived from maps (e.g., traffic speed limits).

The paper "A Real-Time Taxicab Recommendation System Using Big Trajectories Data" by P. Chen et al. proposes a novel algorithmic approach for recommending either a vacant or an occupied taxicab in response to a passenger's request. The recommendation algorithm indicates the closest vacant taxicab to passengers; otherwise, it infers destinations of occupied taxicabs by similarity comparison and clustering algorithms and then recommends to passengers an occupied taxicab heading to a nearby destination.

We do hope that this special issue will be of considerable interest to the Wireless Communications and Mobile Computing's audience, highlighting state-of-the-art trends, methodologies, and applications in smart city environments.

Acknowledgments

We would like to sincerely thank the authors of all the submitted papers for considering our special issue and the Wireless Communications and Mobile Computing as a potential publication venue for their research results. We would also like to especially thank the authors of the accepted papers for their effort in revising and improving their work, occasionally, several times, in response to reviewers' comments. In addition, we would like to thank the anonymous reviewers for doing an excellent job in reviewing the submitted papers and making this special issue possible. Last but not least, we take this opportunity to thank the Editorial Board for giving us the opportunity to organize this special issue, which we sincerely believe provides a fresh, relevant, and useful overview of ongoing research in the multifaceted area of smart cities.

Damianos Gavalas
Petros Nicopolitidis
Achilles Kameas
Christos Goumopoulos
Paolo Bellavista
Lampros Lambrinos
Bin Guo

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 6783240, 11 pages
https://doi.org/10.1155/2017/6783240

Research Article

A Hybrid Service Recommendation Prototype Adapted for the UCWW: A Smart-City Orientation

Haiyang Zhang,[1] Ivan Ganchev,[1,2] Nikola S. Nikolov,[1,3] Zhanlin Ji,[1,4] and Máirtín O'Droma[1]

[1] *Telecommunications Research Centre (TRC), University of Limerick, Limerick, Ireland*
[2] *Department of Computer Systems, University of Plovdiv "Paisii Hilendarski", Plovdiv, Bulgaria*
[3] *Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland*
[4] *North China University of Science and Technology, Tangshan, China*

Correspondence should be addressed to Ivan Ganchev; ivan.ganchev@ul.ie

Received 1 April 2017; Revised 11 August 2017; Accepted 20 August 2017; Published 12 October 2017

Academic Editor: Damianos Gavalas

With the development of ubiquitous computing, recommendation systems have become essential tools in assisting users in discovering services they would find interesting. This process is highly dynamic with an increasing number of services, distributed over networks, bringing the problems of cold start and sparsity for service recommendation to a new level. To alleviate these problems, this paper proposes a hybrid service recommendation prototype utilizing user and item side information, which naturally constitute a heterogeneous information network (HIN) for use in the emerging ubiquitous consumer wireless world (UCWW) wireless communication environment that offers a consumer-centric and network-independent service operation model and allows the accomplishment of a broad range of smart-city scenarios, aiming at providing consumers with the "best" service instances that match their dynamic, contextualized, and personalized requirements and expectations. A layered architecture for the proposed prototype is described. Two recommendation models defined at both global and personalized level are proposed, with model learning based on the Bayesian Personalized Ranking (BPR). A subset of the Yelp dataset is utilized to simulate UCWW data and evaluate the proposed models. Empirical studies show that the proposed recommendation models outperform several widely deployed recommendation approaches.

1. Introduction

With the rapid development of ubiquitous computing, people today are able to access any services anytime and anywhere. Many studies have been done in exploiting wireless communications models for use in ubiquitous network, for example, NGMN (Next Generation Mobile Network) [1] and MUSE (Mobile Ubiquity Service Environment) [2]. Among them, the ubiquitous consumer wireless world (UCWW) [3, 4] brings a different approach to the current global wireless environment, setting out a generic network-independent and consumer-centric techno-business model (CBM) foundation for future wireless communications. The primary change the UCWW brings is that the users become consumers instead of subscribers and thus potentially are able to use the mobile service of any service provider (SP) via the "best" available access network of any access network provider (ANP). Figure 1 depicts a high-level view of the UCWW [3].

One of the key UCWW features is related to the provision of a personalized and customized list of preferred mobile services to consumers by taking into account their preferences as well as the current network and service context [5]. The following are some possible scenarios for utilizing the UCWW within the smart-city paradigm [6]:

(i) Smart parking service: when a consumer in her/his car enters a university/hospital campus or a similar facility, s/he will automatically get a recommendation for the "best" car parking spaces, with allocation and reservation options subject to her/his profile preferences and campus parking policies. The recommendation will come with enhanced functions and information options, if required by the consumer profile,

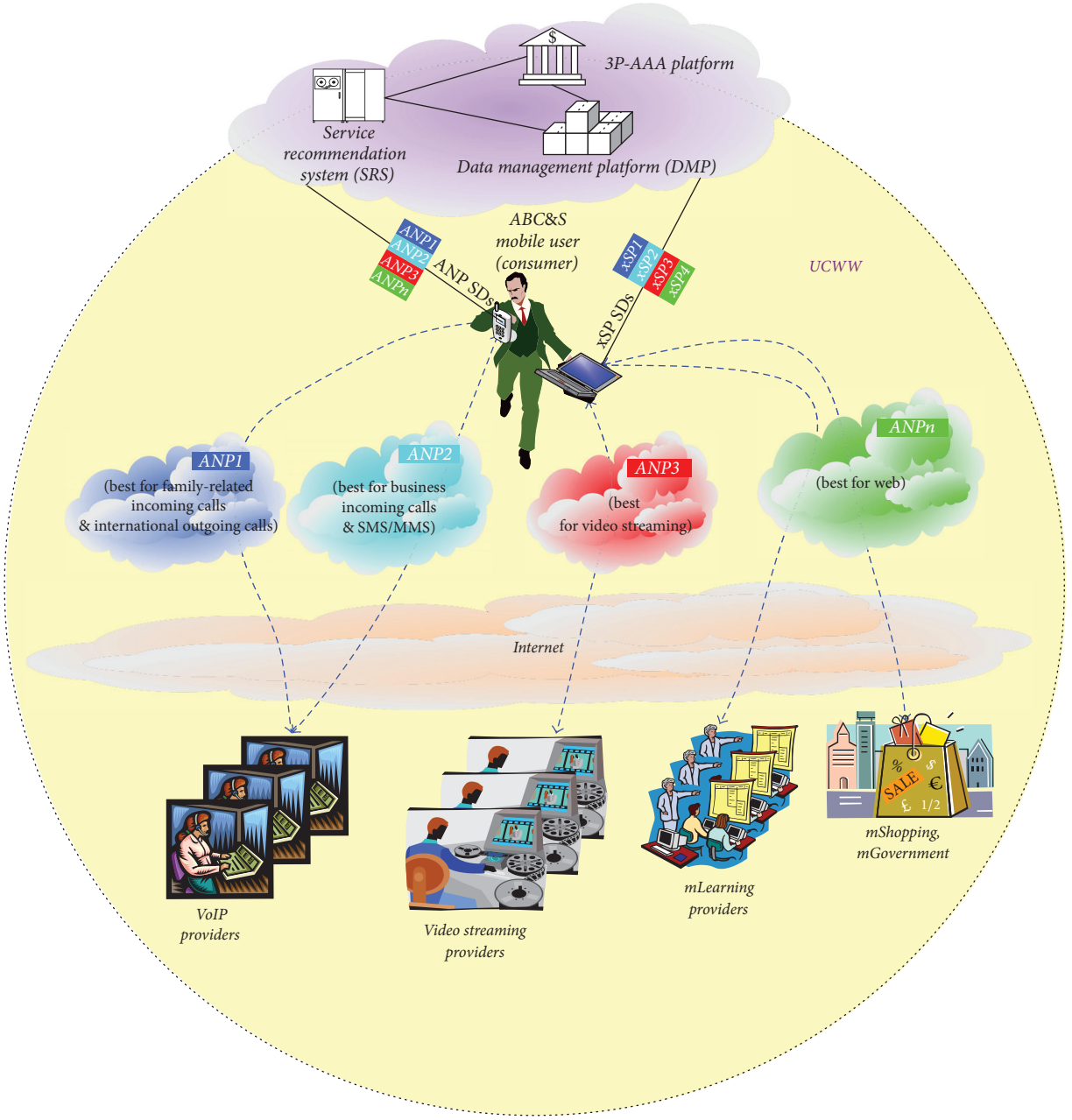


Figure 1: The UCWW: a high-level view.

for example, reservation fee payment scheme and detailed directions to that parking space on a standard navigator app or other proprietary app. Options for provision of all or part of this service, for example, the key parking space reservation, can be made under other conditions, for example, as a "yes" response to "reserve parking at my work-place" pop-up on a mobile device first thing in the morning, even before leaving from home to go to work.

(ii) Personal-health location reminders: the goal of this service is to present the consumer with up-to-date notifications about lowest priced consumer-prescribed drugs in drugstores/pharmacies within the geographic location of the consumer. There would be matching service descriptions (SDs) for apps to collect and collate the information, for example, as part of a cloud-based service recommendation system, from cooperating drugstores. In the SD for such an app, alerts or reminders may be set manually through profile policy, when the consumer is within easy reach of a drugstore with the lowest priced drug. There are many consumer-oriented variations of such a kind of service, leading to many ways

personal-health location reminders may work for different people. Also, this service can potentially support other smart-city healthy living applications, for example, targeted profile-based real-time alerts about areas of high and low pollen count, pollution, air quality index (AQI), and so on or more specific alerts about consumer moves around the city.

In order to support consumer requirements in scenarios such as those described above, recommendation techniques become essential tools assisting consumers in seeking the best available services. The services in the UCWW are divided into two broad categories: *access network communication services* (ANCSs) and *teleservices* (TSs) [7]. ANCSs are used by the consumer to find and use the best access network available in the current location, while TSs are more complex, containing all non-access-network services, from e-learning to online Internet shopping, email, and multimedia services [4]. In this work, we only focus on TSs recommendation problems. The terms "services" and "items" are used to refer to TSs, and "users" is used to refer to consumers in the rest of the paper.

In this paper, a hybrid recommendation prototype for TSs advertising is proposed, working as a platform to assist service providers to reach their valuable targeted users, while at the same time offering each user a list of ranked service instances they may be interested in. To alleviate the cold start and sparsity problems, we propose to leverage the rich side information related to users and services, constructed as a heterogeneous information network (HIN), to build the proposed recommendation models. The proposed models can be potentially also utilized in other recommendation systems. The contributions of this paper are summarized as follows:

(i) First, we design a layered recommendation framework for use in the UCWW, consisting of an offline modeling part and an online recommendation part.

(ii) Second, we propose to leverage HIN to model the information related to users and services, from which rich entity relationships can be generated. The rich relationships are combined with implicit user feedback in a collaborative filtering way to alleviate the cold start and sparsity problems. Recommendation models are defined at both global and personalized level in this paper and are estimated by the Bayesian Personalized Ranking (BPR) optimization technique [8].

(iii) Third, we select a subset of the Yelp dataset to construct the HIN which is complementary to the UCWW service recommendation scenario. Based on this dataset, extensive experimental investigations are conducted to show the effectiveness of the proposed models.

The remainder of the paper is organized as follows. Section 2 presents some related work in this area. Section 3 introduces the background and preliminaries for this study. Section 4 presents the layered configuration of the recommendation prototype architecture. The proposed global and personalized recommendation models are presented in Section 5, with parameters estimated in Section 6. Section 7 presents and analyses the experimental results. Finally, Section 8 concludes the paper and suggests future research directions.

2. Related Work

2.1. Collaborative Filtering with Additional Information. Collaborative filtering (CF) is the most successful and widely used recommendation approach to build recommendation systems. It focuses on learning user preferences by discovering usage patterns from the user–item relations [9]. CF recommendation algorithms are typically favored over content-based filtering (CBF) algorithms due to their overall better performance in predicting common behavior patterns [10]. In the past few decades, huge amount of work was done on exploiting user–item rating matrices to generate recommendations [11–14].

In recent years, there is an increasing trend in exploiting various kinds of additional information to solve the cold start and sparsity problems in CF as well as to improve the recommendation quality of CF models. With the prevalence of social media, social networks have been popular resource to exploit in order to improve recommendation performance. Ma et al. [15] introduce a novel social recommendation framework fusing the user–item matrix with users' social trust networks using probabilistic matrix factorization. Guo et al. [16] propose a trust-based matrix factorization approach, TrustSVD, which takes both implicit influence of ratings and trust into consideration in order to improve the recommendation performance and at the same time to reduce the effect of the data sparsity and cold start problems. User and item side information is also a popular information source for incorporation into CF models in the form of tags [17, 18], user reviews [19, 20], and so on.

To further improve the recommendation performance, HINs have been used to model information related to users and items, in which entities are of various types and links represent various types of relations [21]. Yu et al. [22] introduce a matrix factorization approach with entity similarity regularization, where the similarity is derived from metapaths in a HIN. Luo et al. [23] proposed a social collaborative filtering method, HeteCF, based on heterogeneous social networks. Zheng et al. [24] propose a new dual similarity regularization to enforce the constraints on both similar and dissimilar objects based on a HIN. Majority of the works related to HINs are based on explicit feedback data; few works have been done exploiting implicit feedback data. Yu et al. [25] propose to utilize implicit feedback data to diffuse user preferences along different metapaths in HINs for recommendation generation. However, there are some limitations to this work. Firstly, the authors learn a low-rank representation for the diffused rating matrix under each metapath, which makes the computational complexity of the model training stage relatively high. Secondly, the authors make personalized recommendation based on a group of users obtained by clustering. However, finding a suitable number of clusters for a dataset is a challenging problem and

the recommendation performance heavily depends on the quality of the clusters.

In this study, we propose to use item similarities along different metapaths in a HIN directly to enrich the item-based CF. Recommendation models are defined at both global and personalized level, where different metapath weights are learned for each user avoiding the use of user clusters.

2.2. Top-N Recommendation with Implicit Feedback. Every recommendation algorithm relies on the past user feedback, for example, the user profiling in CBF and the user similarity analysis in CF. The feedback is either explicit (ratings, reviews, etc.) or implicit (clicks, browsing history, etc.) [26]. Although it seems more reliable to make recommendations using the information explicitly supplied by users themselves, the users are usually reluctant to spend extra time or effort on supplying such information, and sometimes the information they provide is inconsistent or incorrect [27]. Compared to explicit feedback, implicit feedback can be collected in a much easier and faster way and at a much larger scale, since it can be tracked automatically without any user effort. For this reason, there has been an increasing research attention to the task of making recommendations by utilizing implicit feedback as opposite to explicit feedback data [28].

Along with recommendation, based on implicit feedback, in the last few years, great attention was paid to the top-N recommendation problem. Many works have been published addressing both tasks [29, 30]. While rating prediction attempts to predict unrated values for each user as accurate as possible, top-N recommendation aims at discovering a ranked list of items which are the most interesting for the user.

In the UCWW recommendation scenario, with consumers feedback available, the proposed hybrid recommendation methods should be able to provide a list of top-N services for each active consumer.

3. Background and Preliminaries

3.1. Heterogeneous Information Network. Most entities in the real world are interconnected, which can be represented with information networks, for example, social networks and research networks. The entity recommendation problem also exists in an information network environment, with items recommended by mining different type of relations from resources that are related to users and items.

In real-world recommendation scenarios, multiple-type objects and multiple-typed links are involved. Thus, the recommendation problem could be modeled with heterogeneous information networks (HINs) [21]. The following definition of an information network was adopted from [21].

Definition 1 (information network). An information network is defined as a directed graph $G = (V, E)$ with an object type mapping function $\phi : V \to A$ and a linked type mapping function $\varphi : E \to R$. Each object $v \in V$ belongs to one particular object type $\phi(v) \in A$, and each link $e \in E$ belongs to a particular relation $\varphi(e) \in R$ [21].

When the number of object types $|A|$ is greater than 1 or the number of relation types $|R|$ is greater than 1, the network is called a *heterogeneous information network (HIN)*; otherwise, it is a *homogeneous information network*.

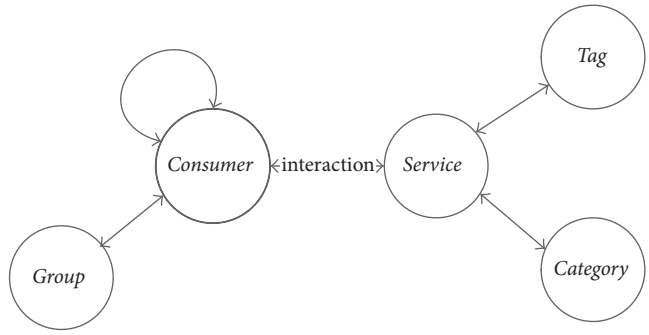


Figure 2: Network schema in UCWW.

In a HIN, an abstract graph is used to represent the entity and relation-type restrictions as per the following definition.

Definition 2 (network schema). A *network schema* S_G of G is the directed graph defined over the object type A with edges from R, denoted as $S_G = (A, R)$, [21].

The definition of the network schema sets the rules on what types of entities exist and how they are connected in an information network. The network schema designed for use in the UCWW service recommendation is shown in Figure 2. Links between a consumer and a service denote their interactions; links between a service and a tag, or a service and a category, denote the corresponding attributes for a service; and links between a consumer and a group, or a consumer and another consumer, denote their social relationships.

In a HIN, two entity types could be connected via different types of relationships following the network schema, thus generating a *metapath*.

Definition 3 (metapath). A metapath $P = A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$ is a path defined on a network schema $S_G = (A, R)$ [21].

Each metapath can be considered as a type of a path in an information network, representing one relation between entity pairs in a HIN. An example of service recommendation in the personal-health location reminder scenario mentioned in Section 1 is described in Example 1.

Example 1. A drug sale reminder service, which advertises a healthcare product, will belong to the "personal-health" category and will have tags like "sale," "healthcare," and so on which are supposed to be defined by the service providers. For a consumer c, if the recommendation system found that some of this consumer's friends used the same service in the last two weeks, this service will be in the rank list for recommendation to consumer c under a *consumer-consumer-service* metapath.

3.2. Metapath Based Similarity. In a HIN, rich similarities between entities can be generated following different

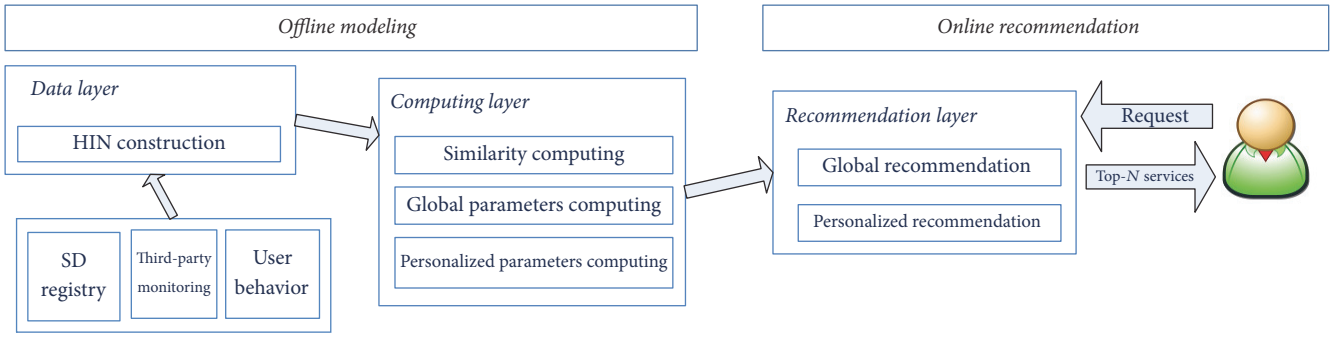


Figure 3: The UCWW service recommendation architecture.

metapaths. Different metapaths represent different semantic meanings; for example, *user-user* denotes social relation between two users and *user-service-user* means that two users are similar because they have similar service-usage histories. The network mining approaches used in homogeneous information networks, such as the random walk used in personalized PageRank [31] and the pairwise random walk used in the SimRank [32], are not suitable for HINs because they are biased to either highly visible or highly concentrated objects [33]. In this study, the *PathSim* approach is utilized to quantitatively measures the same-type objects' similarity in a HIN along symmetric metapath [33]. Given two entities x, y belonging to the same type in a HIN, the *PathSim* is defined as follows [33]:

$$\mathrm{Sim}^{P}(x, y) = \frac{2 \times w(x, y \mid P)}{w(x, x \mid P) + w(y, y \mid P)}, \quad (1)$$

where P denotes the path type and $w(x, y \mid P)$ denotes the number of path instances between x and y along metapath P.

4. UCWW Service Recommendation Architecture

The service recommendation system in the UCWW [34, 35] works as a platform for connecting service providers with consumers. The service recommendation architecture consists of three layers (Figure 3). The data layer and the computing layer belong to the offline modeling part, in which the similarities between services along different metapaths and their corresponding weights are precomputed. In the online recommendation part, the top-N services for the active user are computed at the recommendation layer, based on the results provided by the offline modeling part.

4.1. Data Layer. Information related to users and services is collected and extracted at this layer to construct a HIN, which works as both a service repository and a knowledge base. Compared to most semantic-based recommendation approaches utilizing existing knowledge base or ontology [36], recommendation using a HIN as a knowledge base is more flexible, as it is able to define its own rules (network schema in HIN) for different recommendation requirements.

As shown in Figure 3, in the UCWW, information about consumers and services is collected from three different sources:

(i) A central registry, where service descriptions (SDs) are stored, including attributes such as category, quality of service (QoS), biding price, and consumers package [37]

(ii) A third-party monitoring platform, which provides information about the number of clicks/requests made by consumers for services

(iii) User interactions with services in the past, or social relations between users extracted from other social resources, and so on (details about data collection and data management platform can be found in [34, 35]).

4.2. Computing Layer. In a HIN, items could be similar via different types of relations, which represent different reasons for similarity. Therefore, similarity between items in a HIN could be computed from a combination of different relations rather than only from the rating distributions as in the traditional item-based CF. The main task of this layer is to compute service similarities along different metapaths in the HIN and learn the weights for each metapath in both global and personalized recommendation models.

4.3. Recommendation Layer. This is the most external user-facing layer, presenting system facade to the consumers. All the queries are performed through this layer. When a user has a request for finding the "best" instance of a particular service, a ranked list (computed according to a certain recommendation model) is provided as a response back to him/her.

5. Semantic Recommendation Model

In the UCWW recommendation scenario, the number of services and consumers is relatively high, which can cause even more serious cold start and sparsity problems in service recommendation. In this section, we propose to exploit the side information related to services and consumers to alleviate this problem. The side information is first constructed as a HIN, from which rich service similarities under different

semantics are calculated. The proposed models incorporate these similarities into item-based CF to improve the prediction accuracy. For each user, the recommendation system will first calculate the prediction score for each unrated service and then recommend the top-N services with the highest scores to that user.

5.1. Global Recommendation Model. The item-based CF approach tries to find similar items to the target item, based on their rating pattern. However, with an additional data source related to items and users, items could be similar because of different reasons, based on different features of items. In the UCWW context, within the scope of the HIN, services could be similar due to different reasons via different metapaths. For instance, *service-consumer-service* represents the relation used in the traditional item-based CF, denoting that two services are similar because they are used by a group of consumers, while *service-category-service* means that two services are similar because they share the same category. If one can understand the underlying semantic relations between services and discover services based on rich relations, then potentially more accurate recommendations can be provided to the consumers. Based on this observation and the background knowledge presented in Section 3, a global recommendation model [38] is proposed, which utilizes metapaths with the following format: *service*−∗−*service*.

Given a metapath $P = A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$ with $A_1 = service$ and $A_{l+1} = service$, the predicted value of a consumer c for service i could be defined as follows:

$$r_{c,i} = \sum_{j \in R_c^+} \sum_{p=1}^{L} \theta_p \mathrm{Sim}^p(i, j), \tag{2}$$

where $\mathrm{Sim}^p(i, j)$ is the *PathSim* value between service i and service j along the pth metapath, L is the number of different metapaths considered, θ_p is the weight of the pth metapath among all L metapaths (since different types of metapaths represent different relationship semantics and naturally have different importance in the recommendation model), and R_c^+ denotes the set of services with user interactions in the past.

5.2. Personalized Recommendation Model. With the global recommendation model proposed in the previous subsection, consumers are provided with potentially interesting (for them) services, based on both different types of service relations with rich semantic meanings and service-rating patterns from consumer feedback. However, in real-world UCWW scenarios, consumers' interests in particular features may differ from each other. For instance, taking the online shopping case as an example, the price of a photo camera is usually much more important criterion for buying than its color, which could be learned from the global recommendation model. However, it may happen that one consumer simply wants a camera of a certain color regardless of the price, which means that the metapath, which includes the corresponding tag (a certain color), should have higher importance. In this case, the accuracy of the global recommendation model may not be sufficient because it only considers the overall weights of features without taking into consideration the consumers' individual preferences. In order to better capture the consumer preferences and interests, a fine-grained *personalized recommendation model* is also elaborated in this work, with consideration of every consumer's interests. It allows a higher degree of personalization compared to the global recommendation model. The personalized recommendation model applied to consumer c and service i is defined as follows:

$$r_{c,i} = \sum_{j \in R_c^+} \sum_{p=1}^{L} w_{c,p} \mathrm{Sim}^p(i, j), \tag{3}$$

where $w_{c,p}$ represents the weight of interest of consumer c in the pth feature (metapath) and w_c is the vector representing the consumer's preferences for all features (metapaths).

Compared to the global recommendation model with L parameters to learn, the personalized recommendation model needs to learn $|C| \times L$ parameters, where C is the set of customers.

For both the global and personalized recommendation models, given a consumer, one can calculate the recommendation scores for all services by utilizing either (2) or (3), and then the top-N services can be returned to that consumer as the recommendation result. Parameter estimation methods for both models are introduced in the next section.

6. Recommendation Models Optimization

The objective of the recommendation task is to recommend unrated items with the highest prediction score to each user. A large number of previous studies concentrate on predicting unrated values for each user as accurately as possible. However, the ranking over the items is more important [39]. Considering a typical UCWW recommendation scenario, with only a binary consumer feedback available, a rank-based approach, Bayesian Personalized Ranking (BPR) [8], could be utilized to estimate parameters in the proposed recommendation models. The assumption behind BPR is that the user prefers a consumed item to an unconsumed item, aiming to maximize the following posterior probability:

$$p(\Theta \mid R) \propto p(R \mid \Theta)\, p(\Theta), \tag{4}$$

where R is the rating matrix, $p(\Theta \mid R)$ represents the likelihood of the desired preference structure for all users according to R, and Θ is the parameter vector of an arbitrary model. Thus, BPR is based on pairwise comparisons between a small set of positive items and a very large set of negative items from the users' histories. BPR estimates parameters by minimizing the loss function defined as follows [8]:

$$O = -\sum_{c \in C} \sum_{i \in R_c^+, j \in R_c^-} \ln \sigma\left(r_{c,i} - r_{c,j}\right) + \lambda \|\Theta\|^2, \tag{5}$$

where $\sigma = 1/(1 + e^{-x})$ is the sigmoid function of x, C is the set of available consumers, $r_{c,i}$ and $r_{c,j}$ are the predicted scores of consumer c for items i and j, and R_c^- is the set of items without user ratings yet. Parameters are estimated by means of minimization.

Input: R: implicit feedback
G: information network
Output: Learned global meta-path weights θ
(1) Initialize θ
(2) Generate triples $D_s = \{d((c,i,j) \mid i \in R_c^+,\ j \in R_c^-)\}$
(3) **while** *not converged* **do**
(4) **while** $d \in D_s$ **do**
(5) compute $\partial O/\partial\theta$ with equation (6)
(6) **end**
(7) $\theta \longleftarrow \theta - \alpha \dfrac{\partial O}{\partial \theta}$
(8) **end**

Algorithm 1: Global recommendation model learning.

6.1. Global Recommendation Model Learning. In the global recommendation model, the parameter for estimation is $\theta = \{\theta_1, \ldots, \theta_L\}$ which represents the global weights of all metapaths considered.

The gradient descent (GD) approach [40] could be used to estimate this parameter. The gradient with respect to θ can be calculated as follows:

$$\frac{\partial O}{\partial \theta} = -\sum_{c \in C} \sum_{i \in R_c^+, j \in R_c^-} \frac{e^{-r_{cij}}}{1 + e^{-r_{cij}}} \frac{\partial}{\partial \theta} r_{cij} + \lambda \theta, \tag{6}$$

where $r_{cij} = r_{ci} - r_{cj}$. For each θ_p in $\theta = \{\theta_1, \ldots, \theta_L\}$, the gradient of r_{cij} is

$$\frac{\partial r_{cij}}{\partial \theta_p} = \sum_{k \in R_c^+} \left(\mathrm{Sim}^p(i,k) - \mathrm{Sim}^p(j,k)\right). \tag{7}$$

The process of learning the global recommendation model is presented in Algorithm 1.

6.2. Personalized Recommendation Model Learning. In the personalized recommendation model learning process, one need to learn $|C| \times L$ parameters, with a weighted vector of metapaths for each consumer. Considering the large number of consumers and services in the UCWW and the corresponding huge number of parameters to learn, we employ the stochastic gradient descent (SGD) [41] approach to estimate the parameters for the personalized recommendation model.

Similar to (6), for each triple $(c,i,j) : (c,i) \succ (c,j)$, the update step with respect to $w_{c,p}$ is based on BPR and for each triple it is computed as follows:

$$\frac{\partial O}{\partial w_{c,p}} = -\sum_{c \in C} \sum_{i \in R_c^+, j \in R_c^-} \frac{e^{-r_{cij}}}{1 + e^{-r_{cij}}} \frac{\partial}{\partial w_{c,p}} r_{cij} + \lambda w_{c,p}. \tag{8}$$

For each $w_{c,p}$, the gradient for r_{cij} is estimated as

$$\frac{\partial r_{cij}}{\partial w_{c,p}} = \sum_{k \in R_c^+} \left(\mathrm{Sim}^p(i,k) - \mathrm{Sim}^p(j,k)\right). \tag{9}$$

The learning algorithm for the personalized recommendation model is presented in Algorithm 2.

Table 1: Statistics of the dataset used in the experiments.

Relations $a \leftrightarrow b$	Number of a	Number of b	Number of relations
Consumer ↔ service	2000	5000	8757
Consumer ↔ consumer	2000	2000	2454
Consumer ↔ group	2000	11	9484
Service ↔ category	5000	47	49981
Service ↔ tag	5000	511	14001

Table 2: Metapaths considered in experiments.

Metapath	Notation
consumer-(service-consumer-service)	Pure item-based CF
consumer-(service-consumer-consumer-service)	Consumer social relation enriched item-based CF
consumer-(service-consumer-group-consumer-service)	Consumer group enriched item-based CF
consumer-(service-category-service)	CBF with one feature related to items considered
consumer-(service-tag-service)	
consumer-(service-tag-service-tag-service)	

7. Experiments

7.1. Experiment Setup. In order to simulate a typical UCWW recommendation scenario, we define the network schema for the proposed recommendation prototype as shown in Figure 2. We select a subset of the Yelp dataset (https://www.yelp.ie/dataset_challenge), which contains user ratings on local business and attributes information related to users and businesses. After preprocessing, the new dataset consists of five matrices, representing different relations. The details of the dataset are shown in Table 1. In this dataset, the consumer-service matrix contains 2000 consumers with 8757 service binary interactions on 5000 services, which leads to an extremely sparse matrix with a sparsity of 99.91%.

We randomly take 70% of the consumer-service interaction dataset as a training set and use the remaining 30% as a test set. Six different types of metapaths were utilized for both models in the information network, in the format of *service−∗−service* as shown in Table 2. For BPR parameter estimation, fifty triples $(c,i,j) : (c,i) \succ (c,j)$ are randomly generated for each consumer in the training set.

7.2. Evaluation Metrics and Comparative Approaches. In the proposed service recommendation prototype, a ranked list of services with top-N recommendation score is provided to the consumers. Precision, recall, and $F1$-Measure are used to measure the prediction quality [42]. In the UCWW service recommendation prototype, precision indicates how many services are actually relevant among all selected/recommended services, whereas recall gives the

Input: R: implicit feedback
G: information network
Output: Learned personalized meta-path weight marix W
(1) Initialize W
(2) Generate triples $D_s = \{d((c, i, j) \mid i \in R_c^+, \ j \in R_c^-)\}$
(3) **while** *not converged* **do**
(4) **while** $d \in D_s$ **do**
(5) compute $\partial O/\partial w_{c,p}$ with equation (8)

$$w_{c,p} \longleftarrow w_{c,p} - \alpha \frac{\partial O}{\partial w_{c,p}}$$

(6) **end**
(7) **end**

Algorithm 2: Personalized recommendation model learning.

number of selected/recommended services among all relevant services.

$$\text{precision} = \frac{|\{\text{recommended services}\} \cap \{\text{used services}\}|}{|\{\text{recommended services}\}|},$$
$$\text{recall} = \frac{|\{\text{recommendded services}\} \cap \{\text{used services}\}|}{|\{\text{used services}\}|}. \quad (10)$$

In the evaluation of the adopted top-N recommendation model, precision is normally inversely proportional to recall. When N increases, recall increases as well, whereas precision decreases. Therefore, the $F1$-Measure, which is the harmonic mean of precision and recall [43], was also used as per the following definition:

$$F1\text{-Measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}. \quad (11)$$

For all the three evaluation metrics, a higher score indicates better performance of the corresponding approach.

To demonstrate the effectiveness of the proposed models, we evaluated and compared them with the following widely deployed recommendation approaches:

(i) Item-based CF (IB-CF): this is the traditional and widely used item-based collaborative filtering that recommends items based on the item's k-nearest neighbors [11].

(ii) BPR-SVD: this method learns the low-rank approximation for the user feedback matrix based on the rank of the items, with model learning by BPR optimization technique [8].

We use Hybrid-g and Hybrid-p to denote the proposed global and the personalized recommendation models, respectively.

7.3. Experimental Results. To examine the effectiveness of the proposed recommendation models, we experimentally

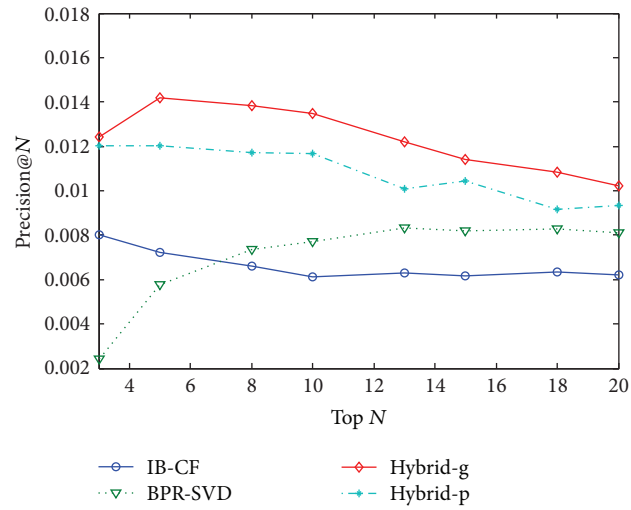


Figure 4: Precision over different N (top-N) values.

computed the top-N list, containing items with the highest top-N recommendation score for each consumer in the test set. The evaluation and comparison results are shown in Figure 4 (precision), Figure 5 (recall), and Table 3 ($F1$-Measure), from which several observations can be drawn.

(i) First, IB-CF outperforms BPR-SVD for small values of N for both precision and recall, but BPR-SVD achieves better results when N increases ($N > 7$).

(ii) Second, the two proposed recommendation models (Hybrid-g and Hybrid-p) sufficiently outperform the other two methods over a wide range of values of N.

(iii) Third, the global model Hybrid-g shows overall better recommendation accuracy than the personalized model Hybrid-p, which may be due to the sparsity of the rating matrix as a relatively small number of rated items cannot truly reflect the true interests of consumers.

Similar to the IB-CF, the rich similarities generated from the HIN in proposed models can be also precomputed and updated periodically offline, as well as the learned weights for

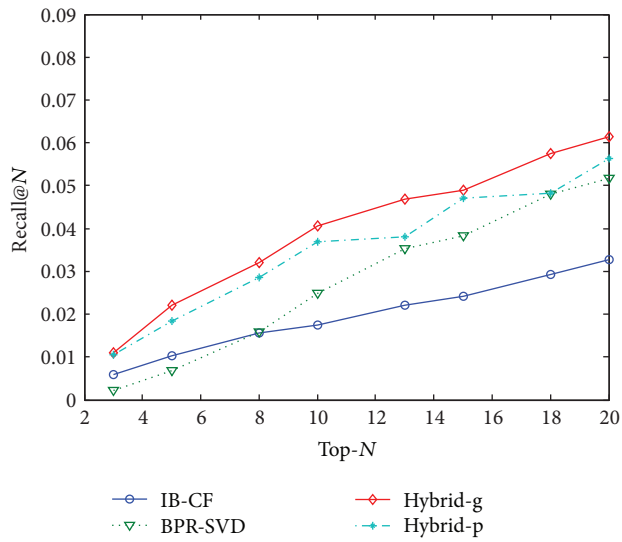


Figure 5: Recall over different N (top-N) values.

Table 3: $F1$-Measure for different N (top-N) values.

$F1$-Measure	$F1@5$	$F1@10$	$F1@15$	$F1@20$
IB-CF	0.00849	0.009085	0.009833	0.010424
BPR-SVD	0.006279	0.011774	0.013485	0.01404
Hybrid-g	**0.017263**	**0.020244**	**0.018482**	**0.017534**
Hybrid-p	0.014582	0.017731	0.017073	0.016001

Table 4: Comparison of computational complexities of compared recommendation algorithms.

Algorithms	Offline	Online
IB-CF	$O(m^2n)$	$O(mh)$
BPR-SVD	$O(tdn)$	$O(md)$
Hybrid-g	$O(m^2n\|L\| + nt)$	$O(mh)$
Hybrid-p	$O((m^2n + nt)\|L\|)$	$O(mh\|L\|)$

both models. Given n consumers and m services, for an active consumer, the upper bound of the computational complexity for top-N recommendation among all algorithms addressed in this paper is shown in Table 4 where h denotes the number of services the active consumers already used, t is the number of iterations for learning parameters and d is the number of latent features in the matrix factorization approach, and $|L|$ is the number of metapaths considered in the proposed models.

As h and d are much smaller than m, we can assume that the proposed global recommendation model has similar computational complexity to both the traditional IB-CF and BPR-SVD approaches in the online recommendation stage but higher computational complexity in the offline modeling stage for achieving better effectiveness. The computational complexity of the personalized recommendation model is higher than the global recommendation model in both the offline and online stages, with a different set of weights for each user to learn and combine. Between the proposed models, the global recommendation model provides better results than the personalized model and achieves this with lower computation complexity in both the offline modeling stage and the online recommendation stage.

8. Conclusion

Mobile phones are currently the most popular personal communication devices. They have formed a new media platform for merchants with their anytime-anywhere accessible functionalities. However, the most important problem for merchants is how to deliver a service to the right mobile user in the right context efficiently and effectively. The proposed service recommendation prototype can potentially provide a platform to assist service providers to reach their valuable targeted consumers.

The integration of the proposed service recommendation system prototype into the ubiquitous consumer wireless world (UCWW) has the potential to create an infrastructure in which consumers will have access to mobile services, including those supporting smart-cities operation, with a radically improved contextualization. As a consequence, this environment is expected to radically empower individual consumers in their decision making and thus positively impact the society as a whole. It will also facilitate and enable a direct relationship between consumers and service providers. Such direct relationship is attractive for the effective development of smart-city services since it allows for more dynamic adaptability and holds the potential for user-driven service evolution. Besides benefiting consumers, the UCWW opens up the opportunity for stronger competition between service providers, therefore creating a more liberal, more open, and fairer marketplace for existing and new service providers. In such a marketplace, service providers can deliver a new level of services which are both much more specialized and reaching a much larger number of mobile users.

The recommendation prototype proposed in this paper could be potentially employed for discovering the "best" service instances available for use to a consumer through the "best" access network (provider), realizing a consumer-centric always best connected and best served (ABC&S) experience in UCWW. In line with the layered architecture of the service recommendation prototype, two hybrid recommendation models which leverage a heterogeneous information network (HIN) are proposed at a global and personalized level, respectively, for exploiting sparse implicit data. An empirical study has shown the effectiveness and efficiency of the proposed approaches, compared to two widely employed approaches. The proposed recommendation models also have the potential to work under other recommendation scenarios effectively.

However, for service recommendation in the UCWW, we only provided the basic recommendation models in this paper, without considering real-time context information. Also, the similarity matrices computed from different metapaths are still sparse, which may cause some inaccurate rating predictions. As a future work, we intend to conduct further study on context aware recommendations with a real application operating with big data. We also intend to explore the study of matrix factorization approach on similarity matrices derived from different metapaths.

Disclosure

This paper is extended from the paper entitled "Hybrid Recommendation for Sparse Rating Matrix: A Heterogeneous Information Network Approach," presented at the IAEAC 2017.

Conflicts of Interest

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

Acknowledgments

This publication has been supported by the Chinese Scholarship Council (CSC), the Telecommunications Research Centre (TRC), University of Limerick, Ireland, and the NPD of the University of Plovdiv, Bulgaria, under Grant no. ФП17-ФМИ-008.

References

[1] North Alliance, "NGMN 5G white paper," 2015, https://www.ngmn.org.

[2] J. Yang, Z. Ping, H. Zheng, W. Xu, L. Yinong, and T. Xiaosheng, "Towards mobile ubiquitous service environment," *Wireless Personal Communications*, vol. 38, no. 1, pp. 67–78, 2006.

[3] M. O'Droma and I. Ganchev, "Toward a ubiquitous consumer wireless world," *IEEE Wireless Communications*, vol. 14, no. 1, pp. 52–63, 2007.

[4] M. O'Droma and I. Ganchev, "The creation of a ubiquitous consumer wireless world through strategic ITU-T standardization," *IEEE Communications Magazine*, vol. 48, no. 10, pp. 158–165, 2010.

[5] I. Ganchev, M. O'Droma, N. S. Nikolov, and Z. Ji, "A ucww cloud-based system for increased service contextualization in future wireless networks," in *Proceedings of the 2nd international conference on telecommunications and remote sensing, ICTRS 2013*.

[6] H. Zhang, I. Ganchev, N. S. Nikolov, and M. O'Droma, "A service recommendation model for the Ubiquitous Consumer Wireless World," in *Proceedings of the 2016 IEEE 8th International Conference on Intelligent Systems (IS)*, pp. 290–294, Sofia, Bulgaria, September 2016.

[7] P. Flynn, I. Ganchev, and M. O'Droma, "WBCs -ADA Vehicle and Infrastructural Support in a UCWW," in *Proceedings of the 2006 IEEE Tenth International Symposium on Consumer Electronics*, pp. 1–6, June 2006.

[8] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proceedings of the 25th conference on uncertainty in artificial intelligence*, pp. 452–461, AUAI Press, 2009.

[9] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges," *ACM Computing Surveys*, vol. 47, no. 1, Article ID 2556270, pp. 3:1–3:45, 2014.

[10] R. Ronen, N. Koenigstein, E. Ziklik, and N. Nice, "Selecting content-based features for collaborative filtering recommenders," in *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*, pp. 407–410, Hong Kong, October 2013.

[11] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[12] M. H. Aghdam, M. Analoui, and P. Kabiri, "Collaborative filtering using non-negative matrix factorisation," *Journal of Information Science*, vol. 43, no. 4, pp. 567–579, 2017.

[13] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[14] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08)*, pp. 426–434, New York, NY, USA, August 2008.

[15] H. Ma, H. Yang, and M. R. Lyu, "Sorec: social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*, pp. 931–940, Napa Valley, Calif, USA, October 2008.

[16] G. Guo, J. Zhang, and N. Yorke-Smith, "A novel recommendation model regularized with user trust and item ratings," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1607–1620, 2016.

[17] M. G. Manzato, "gsvd++: supporting implicit feedback on recommender systems with metadata awareness," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, pp. 908–913, Coimbra, Portugal, March 2013.

[18] I. Fernández-Tobas and I. Cantador, "Exploiting social tags in matrix factorization models for cross-domain collaborative filtering," in *Proceedings of the International Workshop on New Trends in Content based Recommender Systems*, pp. 34–41, 2014.

[19] Y. Bao, H. Fang, and J. Zhang, "Topicmf: Simultaneously exploiting ratings and reviews for recommendation," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, vol. 14, pp. 2–8, 2014.

[20] K. Bauman, B. Liu, and A. Tuzhilin, "Recommending items with conditions enhancing user experiences based on sentiment analysis of reviews," in *Proceedings of the International Workshop on New Trends in Content based Recommender Systems*, pp. 19–22, 2016.

[21] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. S. Yu, "A survey of heterogeneous information network analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 1, pp. 17–37, 2017.

[22] X. Yu, X. Ren, Q. Gu, Y. Sun, and J. Han, "Collaborative filtering with entity similarity regularization in heterogeneous information networks," in *Proceedings of the International Joint Conference on Artificial Intelligence workshop on Heterogeneous Information Network Analysis*, vol. 27, 2013.

[23] C. Luo, W. Pang, Z. Wang, and C. Lin, "Hete-CF: Social-Based Collaborative Filtering Recommendation Using Heterogeneous Relations," in *Proceedings of the 2014 IEEE International Conference on Data Mining (ICDM)*, pp. 917–922, Shenzhen, China, December 2014.

[24] J. Zheng, J. Liu, C. Shi, F. Zhuang, J. Li, and B. Wu, "Recommendation in heterogeneous information network via dual similarity regularization," *International Journal of Data Science and Analytics*, vol. 3, no. 1, pp. 35–48, 2017.

[25] X. Yu, X. Ren, Y. Sun et al., "Personalized entity recommendation: A heterogeneous information network approach," in *Proceedings of the 7th ACM international conference on Web search and data mining*, pp. 283–292, New York, NY, USA, Feburary 2014.

[26] M. R. Ghorab, D. Zhou, A. O'Connor, and V. Wade, "Personalised information retrieval: survey and classification," *User Modelling and User-Adapted Interaction*, vol. 23, no. 4, pp. 381–443, 2013.

[27] A. Demiriz, "Enhancing product recommender systems on sparse binary data," *Data Mining and Knowledge Discovery*, vol. 9, no. 2, pp. 147–170, 2004.

[28] Y. Hu, C. Volinsky, and Y. Koren, "Collaborative filtering for implicit feedback datasets," in *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM '08)*, pp. 263–272, IEEE, Pisa, Italy, December 2008.

[29] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-N recommendation tasks," in *Proceedings of the 4th ACM Recommender Systems Conference (RecSys '10)*, pp. 39–46, New York, NY, USA, September 2010.

[30] V. C. Ostuni, T. Di Noia, E. Di Sciascio, and R. Mirizzi, "Top-n recommendations from implicit feedback leveraging linked open data," in *Proceedings of the 7th ACM Conference on Recommender Systems*, pp. 85–92, ACM, New York, NY, USA, 2013.

[31] L. Page, S. Brin, R. Motwani, and T. Winograd, "PageRank citation ranking: bringing order to the web," Stanford InfoLab, 1999.

[32] G. Jeh and J. Widom, "Simrank: a measure of structural-context similarity," in *Proceedings of the 8th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 538–543, Edmonton, Alberta, Canada, July 2002.

[33] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," in *Proceedings of the VLDB Endowment*, vol. 4, pp. 992–1003, 2011.

[34] I. Ganchev, Z. Ji, and M. O'Droma, "A distributed cloud-based service recommendation system," in *Proceedings of the 2015 International Conference on Computing and Network Communications (CoCoNet)*, pp. 212–215, Trivandrum, December 2015.

[35] I. Ganchev, Z. Ji, and M. O'Droma, "A data management platform for recommending services to consumers in the UCWW," in *Proceedings of the 2016 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 405-406, Las Vegas, NV, January 2016.

[36] S. E. Middleton, D. De Roure, and N. R. Shadbolt, "Ontology-Based Recommender Systems," in *Handbook on Ontologies*, pp. 779–796, Springer, Berlin, Heidelberg, 2009.

[37] Z. Ji, I. Ganchev, and M. O'Droma, "Advertisement data management and application design in WBCs," *Journal of Software*, vol. 6, no. 6, pp. 1001–1008, 2011.

[38] H. Zhang, I. Ganchev, N. S. Nikolov, Z. Ji, and M. ODroma, "Hybrid recommendation for sparse rating matrix: A heterogeneous information network approach," in *Proceedings of the in 2017 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2017.

[39] J. Pessiot, T. Truong, N. Usunier, M. Amini, and P. Gallinari, "Learning to rank for collaborative filtering," in *Proceedings of the 9th International Conference on Enterprise Information Systems*, pp. 145–151, Citeseer, Funchal, Madeira, Portugal, 2007.

[40] C. Burges, T. Shaked, E. Renshaw et al., "Learning to rank using gradient descent," in *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, pp. 89–96, ACM, New York, NY, USA, 2005.

[41] M. Zinkevich, M. Weimer, L. Li, and A. J. Smola, "Parallelized Stochastic Gradient Descent," in *Advances in neural information processing systems*, pp. 2595–2603, 2010.

[42] D. M. W. Powers, "Evaluation: from precision, recall and f-measure to roc., informedness, markedness correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[43] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 7391982, 15 pages
https://doi.org/10.1155/2017/7391982

Research Article

Unchained Cellular Obfuscation Areas for Location Privacy in Continuous Location-Based Service Queries

Jia-Ning Luo[1] and Ming-Hour Yang[2]

[1]*Department of Information and Telecommunications Engineering, Ming Chuan University, Taoyuan, Taiwan*
[2]*Department of Information and Computer Engineering, Chung Yuan Christian University, Taoyuan, Taiwan*

Correspondence should be addressed to Ming-Hour Yang; mhyang@cycu.edu.tw

Received 9 February 2017; Revised 6 July 2017; Accepted 10 August 2017; Published 28 September 2017

Academic Editor: Christos Goumopoulos

To access location-based service (LBS) and query surrounding points of interest (POIs), smartphone users typically use built-in positioning functions of their phones when traveling at unfamiliar places. However, when a query is submitted, personal information may be leaked when they provide their real location. Current LBS privacy protection schemes fail to simultaneously consider real map conditions and continuous querying, and they cannot guarantee privacy protection when the obfuscation algorithm is known. To provide users with secure and effective LBSs, we developed an unchained regional privacy protection method that combines query logs and chained cellular obfuscation areas. It adopts a multiuser anonymizer architecture to prevent attackers from predicting user travel routes by using background information derived from maps (e.g., traffic speed limits). The proposed scheme is completely transparent to users when performing continuous location-based queries, and it combines the method with actual road maps to generate unchained obfuscation areas that conceal the actual locations of users. In addition to using a caching approach to enhance performance, the proposed scheme also considers popular tourist POIs to enhance the cache data hit ratio and query performance.

1. Introduction

Currently, most mobile devices feature built-in positioning functions, and smartphone users frequently use location-based services (LBS) to query points of interest (POIs) within their vicinity (e.g., when searching for Chinese restaurants within a 10 km radius). Although using LBSs to rapidly locate places and routes is highly convenient, LBS providers may exploit the opportunity to collect the query contents and travel routes of specific users and then analyze these datasets to determine the users' dietary habits, shopping preferences, and even personal medical histories. These behaviours are a severe breach of LBS user' right to privacy.

Numerous previous scholars [1, 2] developed peer-to-peer (P2P) cloaking algorithms to mask the identity and location of users to guarantee location privacy. These P2P algorithms satisfy k-anonymity by sharing the user location with other users. However, the approaches proposed in those studies search for $k - 1$ conspirators surrounding the user, which may enable attackers to triangulate a user within an obfuscation area (OA) and deploy a variance-based attack (VBA) [3]. In [3], an approach was proposed that searches for other conspirators surrounding a user. Subsequently, a random conspirator in the group is selected to search for other conspirators. This process is repeated until the k-anonymity requirement is satisfied; that is, the user cannot be triangulated within an obfuscation area. Subsequently, P2P necessitates the exchange of location information between users. Therefore, users are required to trust other users in the obfuscation area. A malicious user could select different k to obtain the location of the other users by using the k-anonymity algorithm in [3]. They may even partner with LBS providers to steal personal data from regular users, increasing the risk of privacy leaks.

Recent studies have proposed methods for masking the identity [4, 5], location [6–8], and query information [9] of users by using secure third-party anonymizers to encode the location of a user or POI. Anonymizers not only protect

user privacy but also reduce communication time and costs. One study [4] proposed using an anonymizer to mask the identities of a group of query users by using the identity of one random user in the group. These queries, which contain the same metadata, are transmitted to the LBS server. Another study [7] used a Hilbert curve to create an obfuscation area to mask the location of users. Anonymizers mask users by randomly selecting a representative user in proximity to a group of users. The metadata of the representative are copied to all queries before they are transmitted to the LBS server, thereby satisfying k-anonymity and obfuscation requirements. Anonymizers typically create obfuscation areas in grid [7, 10–12] or pyramid structure to mask user locations. In [13], a method was proposed to resolve the incompatibility between the original obfuscation area and query criteria by creating an additional obfuscated query area to keep privacy.

Even when a user is masked within an obfuscation area to satisfy k-anonymity, LBS servers can collect user queries for area information when they submit continuous LBS queries in a short period. Users are more likely to use LBSs in unfamiliar rural tourist locations (rather than in urban areas) where roads are more dispersed. The simpler road network structures of rural areas enable LBS servers to determine the locations of users by analyzing maps and road conditions. To prevent LBS servers from cross-referencing continuous queries to obtain user location information, previous scholars have added reachable query routes to confuse LBS servers [14–18]. However, LBS servers can extrapolate known data, such as user habits, interests, and actual maps, to determine the most probable route of travel through an elimination process. In [19], a method was proposed to determine reasonable POIs within a user's query area by analyzing his or her past query records. Subsequently, the user's actual location is combined with a corresponding reasonable POI to generate a dummy query, preventing LBS servers from filtering out unreasonable dummy queries. A subsequent study [20] proposed a method that selects a nearby insensitive location from a user's past travel routes to substitute sensitive query locations. However, this method was prone to leak the query location because it failed to account for map data and user mobility. In response, another study [17] combined an anonymizer with map data (all intersection branches within the road network) and user mobility. To confuse LBS servers, the anonymizer used in that study generated obfuscation areas that include the section of road extending from the user's current intersection, but they do not include blind alleys or overlapping routes according to the user's privacy requirements.

In this study, we proposed a method combining the anonymizer provided by trusted third-party servers with actual road maps and users' movement patterns to create multiple virtual paths. When user content cannot be detected in the cache, the mechanism is applied to guarantee the privacy of user queries. The proposed method provides users with high query performance when the query volume is high while guaranteeing location privacy. The method uses the popular query characteristics of tourist locations to enhance the cache hit ratio, query performance, and protection of users' POI and query locations. The proposed method also considers similarities between pseudoqueries and users' actual queries, as well as cached POIs, to prevent the generated pseudoqueries from being filtered out by the LBS server, thereby increasing the cache life and hit ratio. The proposed method in the present study is suitable for continuous queries. It has the following contributions:

(1) The privacy of users' POIs is maintained, even during continuous querying.

(2) POIs that are difficult for LBS servers to filter out are generated by incorporating area characteristics, logs, and user queries.

(3) User privacy requirements are satisfied, even when the location obfuscation algorithm is known to the attacker.

(4) Obfuscation areas are generated from real-time maps, thereby avoiding exposing user locations.

(5) Cache data are used to reduce the communication costs and time of the anonymizer and LBS server.

The remainder of this paper is organized as follows. Section 2 describes the system architecture and initialization phase, and Section 3 discusses the development of the proposed method. The security analysis and the performance analysis are discussed in Section 4, and Section 5 presents the conclusion.

2. System Architecture

The system architecture is illustrated in Figure 1. When multiple users access LBSs to submit queries, the queries are transmitted to a trusted obfuscated server to protect user privacy. An anonymizer cross-references the query content with the cache database. If POI data matching the query content are detected, the query results are encrypted and returned to the users. In Figure 1, the queries "night market" and "super market" are returned to users from the anonymizer (indicated by the dotted line). If relevant data are not cached, the anonymizer obfuscates the user's query and location and transmits the obfuscated query to the LBS server. In Figure 1, the user's "fast food" query and location are obfuscated and transmitted to the LBS server (indicated by the solid line). Once the anonymizer receives the POIs within the obfuscation area from the LBS server, it updates the cache database, filters out the pseudodata, encrypts the query results, and returns the results to the user.

To reduce the computation load of the LBS server for processing user queries, the proposed method uses cell numbers instead of coordinates to represent the query range sent to the LBS server. However, this process necessitates additional computations and transmission costs to synchronize the maps, cell sizes, and cell numbers on the anonymizer and LBS server. Accordingly, we adopted a numbering system for the cellular structure to reduce the overhead costs. This method synchronizes only the center point of the map and the sides of the cells to maintain consistent map segregation and numbering between the anonymizer and LBS server.

The proposed method adopts a trusted anonymizer to protect users' queries from being collected by the LBS server

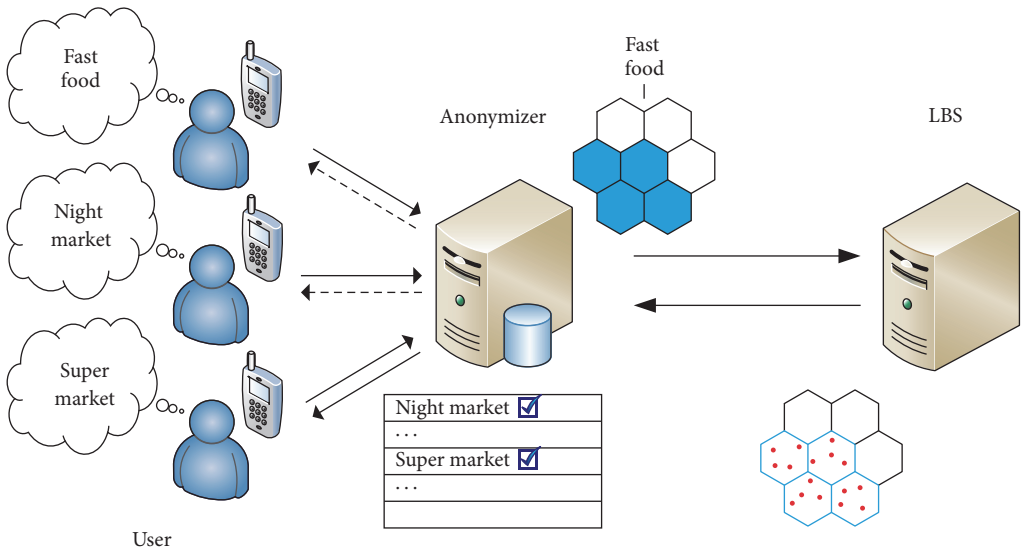


Figure 1: System architecture.

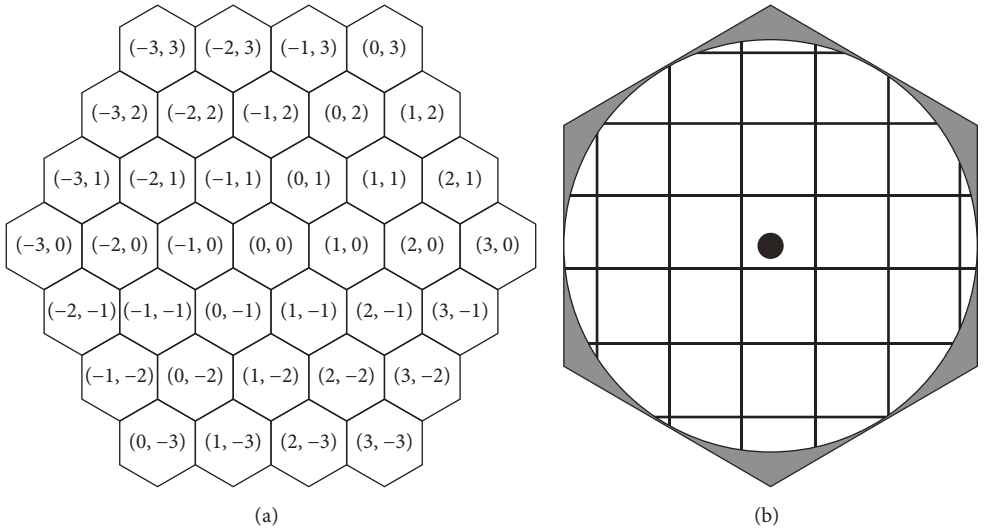


Figure 2: Cellular structure; (a) cell numbering; (b) query range.

or other attackers. However, five criteria must be met to successfully implement the proposed method. First, the map on the LBS server must be divided into a cellular structure with a cell side length R (Figure 2(a)), and the user's query range must be an inscribed circle of the cellular structure (Figure 2(b)), where the query radius is the POI within the range of $(\sqrt{3}/2)R$. Second, the LBS server cannot frequently revise the cellular structure of the map. Third, the anonymizer must be reliable for masking user locations. Fourth, the maps on the anonymizer must contain intersections, length of road sections, and speed-limit information. Fifth, the algorithm must be available to the public.

Threat models for the LBS server and general attackers are defined in this section. The effectiveness of the proposed method for guarding against these threat models is discussed in Section 4. First, the LBS servers and general attackers can continuously tap, collect, and leak user information. However, they do not alter inbound or outbound query information (e.g., query number). Second, attackers use the open obfuscation algorithm and their background knowledge on known intersections, road sections, and traffic speed limits to deduce users' travel routes and determine their locations. Third, query results are returned from the LBS server to the anonymizer. This creates the opportunity for the LBS server or general attackers to analyze the cache data of the anonymizer by using known cache algorithms. Fourth, the LBS server and general attackers can cross-reference the obfuscation areas queried by different user IDs

in different locations and at different times to identify the associations between different queries and determine the query information submitted by the same user.

Without changing the center and side lengths of the cells, the initialization of the anonymizer and LBS server needs to be performed only once (procedures are presented in Section 2.1). We developed a three-phase unchained location privacy protection method for processing user queries (procedures are presented in Section 3). The following section provides the initialization model. Notations lists and explains the notations used in this paper.

2.1. System Initiation. Once the anonymizer obtains the center coordinates $(H_x^{(0,0)}, H_Y^{(0,0)})$ from the LBS server, it uses these coordinates to number each cell and determine their center points. The term $(H_x^{(i,j)}, H_y^{(i,j)})$ denotes the center coordinates of each cell, where (i, j) represents the x- and y-axes of the cell. The cellular-structure map illustrated in Figure 2(a) is used to generate a cellular structure comprising cells with $1 + 3n(n + 1)$ side lengths = R, where n represents the number of layers in the structure. The cells are numbered according to the (i, j) order, where the center of the map is $(0, 0)$. Assuming that the hexagonal cell has six directions, i increases in increments of 1 to the right and decreases in increments of 1 to the left; j increases in increments of 1 to the upper right and decreases in increments of 1 to the bottom left; i increases and j decreases in increments of 1 to the bottom right; and i decreases and j increases in increments of 1 to the upper left. Results are illustrated in Figure 2(a).

Once all the cells in the anonymizer are numbered, set V (all intersection in $G = (V, E)$, which is a figure containing section length weights) and set E with all sections linking two intersections in $G = (V, E)$ are matched to each cell.

$$\begin{aligned}&\left\{V_{(i,j)} \subseteq V \mid V_{(i,j)} = V_{(i,j)}^1 \cup V_{(i,j)}^2 \cup \cdots \cup V_{(i,j)}^6,\ \left|V_{(i,j)}\right| \neq \emptyset\right\},\\&\left\{E_{(i,j)} \subseteq E \mid E_{(i,j)} = E_{(i,j)}^1 \cup E_{(i,j)}^2 \cup \cdots \cup E_{(i,j)}^6,\ \left|E_{(i,j)}\right| \neq \emptyset\right\},\end{aligned} \tag{1}$$

where $V_{(i,j)}^{\mathrm{Tir}}$ represents the intersections contained in triangle Tir in cell (i, j) and $E_{(i,j)}^{\mathrm{Tir}}$ represents the sections with length weights contained in the triangle Tir in cell (i, j).

The numbering method for triangle Tir is illustrated in Figure 3. Tir = 1, 2, 3, 4, 5, and 6 refer to $\Delta P^1P^2P^7$, $\Delta P^2P^3P^7$, $\Delta P^3P^4P^7$, $\Delta P^4P^5P^7$, $\Delta P^5P^6P^7$, and $\Delta P^6P^1P^7$, respectively. This method enables the fewest cells in the obfuscation area to be used to cover the query range. Details concerning the generation procedures and verification of the obfuscation areas are presented in Section 3.

3. Unchained Location Protection Scheme

When a user submits a query, he transmits his ID, $P^a(L_x, L_y)$, POI for the query, and the k-anonymity requirements to the anonymizer. The anonymizer applies the three-phase obfuscation algorithm (Figure 4) to obfuscate his location prior to sending the query to the LBS server. The server then returns the queried information to the anonymizer, which filters out nonuser information before returning the POIs to the user. In Phase 1, the user's real coordinates are used to calculate the cell number of the user location and the triangle Tir within the cell. If the cell number and POI information are already cached in the anonymizer, the algorithm skips to Phase 3. Otherwise, it continues to the next phase. In Phase 2, multiple obfuscation areas are generated according to the user's privacy requirements. The obfuscation area that contains the query range is substituted with a pseudo-ID and a pseudoquery order before it is transmitted to the LBS server. The anonymizer then caches the information returned by the LBS server (including the user's original query and generated pseudoquery). This information can then be used for similar queries in the future. Finally, the anonymizer uses the substituted user ID to retrieve the POI results. In Phase 3, the filtered query results are returned to the user.

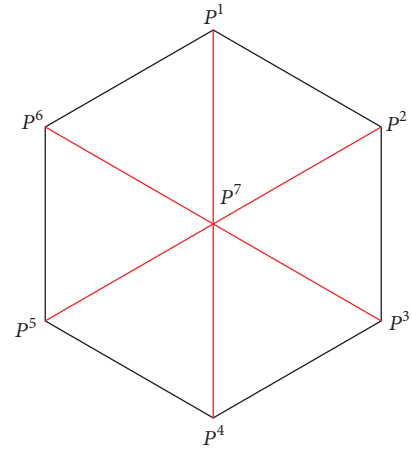


Figure 3: A cell divided into 6 equilateral triangles.

Calculation of the cell numbers is explained in Section 3.1, generation of users' obfuscation areas is described in Section 3.2, generation of multiple pseudoobfuscation areas to protect the privacy of multiple users simultaneously submitting queries and using the cache to achieve unchained location protection are presented in Section 3.3, and query submission is outlined in Section 3.4.

3.1. Calculating User Cell Number. Once the anonymizer receives the current location coordinates of the user $(P^a(L_x, L_y))$, it applies (2) to calculate the vertical displacement of the user relative to the center coordinates of the map $(H_x^{(0,0)}, H_y^{(0,0)})$. The anonymizer then incorporates the vertical displacement into (3) to determine the cell number (i, j) of the user. The calculation process is discussed as follows:

$$j = \begin{cases} 2\left\lfloor \dfrac{\left|L_y - H_y^{(0,0)}\right| + d}{3R} \right\rfloor + e, & \text{where } L_y \geq H_y^{(0,0)} \\ -2\left\lfloor \dfrac{\left|L_y - H_y^{(0,0)}\right| + d}{3R} \right\rfloor - e, & \text{where } L_y < H_y^{(0,0)} \end{cases} \tag{2}$$

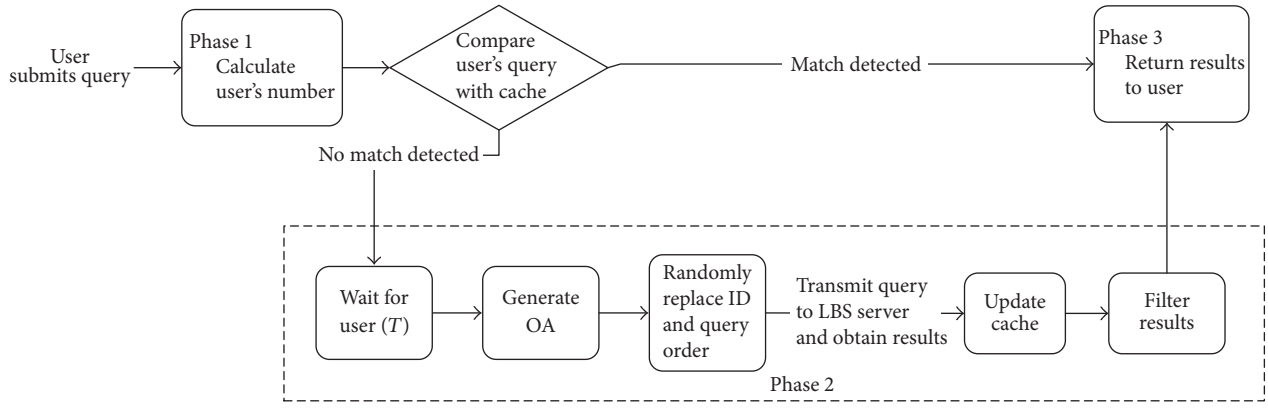


Figure 4: Query process.

$$e = \begin{cases} 1 & \text{if } \dfrac{\left(\left|L_y - H_y^{(0,0)}\right| + d\right) \bmod 3R}{2d} > 1 \\ 0 & \text{if } \dfrac{\left(\left|L_y - H_y^{(0,0)}\right| + d\right) \bmod 3R}{2d} \leq 1. \end{cases} \tag{3}$$

First, the vertical distance between the user (L_y) and origin ($H_y^{(0,0)}$) is used to calculate j of the cell number for the user location. The distance between the center points of two vertically adjacent cells (e.g., (0, 0) and (−1, 2) in Figure 5(a)) is $3R$, and the cell numbers of these two cells differ by 2. When L_y is located above the center coordinates ($L_y \geq H_y^{(0,0)}$), $(L_x, H_y^{(0,0)})$ on $y = H_y^{(0,0)}$ is added to $|L_y - H_y^{(0,0)}|$. This point extends downward vertically to the cell boundary for distance d. The number of cells within d is calculated and incorporated into (3) to determine whether the final section smaller than $3R$ crosses over to another cell. In Example 1 (Figure 5(b)), the user is located on a random point (P^a) of (L_x, L_y) above the center coordinates. $|L_y - H_y^{(0,0)}| + d$ can be expressed as $\overline{P^aP^c} = \overline{P^aP^b} + \overline{P^bP^c} = 3R + \overline{P^aP^d}$ and $j = 2 + e$ (see (2)). Moreover, $e = 0$ because $\overline{P^aP^b} < 2d$ (see (3)). Thus, the user in Example 1 is located in $j = 2$ on the y-axis. In Example 3 (Figure 5(c)), $j = 0 + e$ because $\overline{P^aP^c} < 3R$ (see (2)). However, $e = 1$ because $\overline{P^aP^c} > 2d$ (see (3)). Thus, the user in Example 2 is located in $j = 1$. The cell number (j) of users can be determined using (2) and (3).

The preceding discussion indicates that d must be determined before j is calculated. The length of d is associated with a user's L_x coordinate. The value of d gradually increases as L_x shifts from the cell boundary ($d = R/2$) to the center of the cell ($d = R$). Figure 6 shows that the center distance between two cells is $2R * \cos 30^\circ = \sqrt{3}R$. In other words, one length cycle of d is $\sqrt{3}R$.

$$d = f(L_x) = \frac{1}{\sqrt{3}} \left| \alpha(L_x) - \frac{\sqrt{3}}{2}R \right| + \frac{R}{2} \tag{4}$$

$$\alpha(L_x) = \left|L_x - H_x^{(0,0)}\right| \bmod \sqrt{3}R \tag{5}$$

The term d can be expressed as a triangular wave by using $R \leq d \leq R/2$ and a length cycle of $\sqrt{3}R$. The wave equation is expressed in (4), where $1/\sqrt{3}$ is the positive slope of $0 \leq x \leq \sqrt{3}R$ and $R/2$ is the vertical displacement distance. Because each $\sqrt{3}R$ represents one cycle (see (5)), the relationship diagram between L_x and d can be illustrated by calculating the location of L_x in a single cycle, as shown in Figure 7.

Coordinates on the y-axis increase by $(\sqrt{3}/2)R$ when 1 is added to the y-axis. Therefore, the offset $(j * (\sqrt{3}/2)R)$ caused by the change in the Y-coordinates must be subtracted when calculating the X-coordinates. Then, the current distance between L_x and the origin on the x-axis is multiplied by a unit length to obtain i on the x-axis, as illustrated in Figure 5(b). We know that (L_x, L_y) is located in $(i, 2)$, $j = 2$, and the distance between $H_x^{(0,0)}$ and L_x is less than $(\sqrt{3}/2)R$. Hence, a user cell number of $i = -1$ can be calculated using (3). Therefore, when the location of the user is known (L_x, L_y), $(H_x^{(0,0)}, H_y^{(0,0)})$ and (2) and (3) can be used to determine the current location of the user, that is, (−1, 2) in Figure 5(b).

3.2. Determining the Obfuscation Area. Once the cell number containing the user location and center coordinates is confirmed and if the user query information has not been cached, the anonymizer produces an obfuscation area for the user query and transmits the obfuscation area to the LBS server. The triangle Tir of the cell in which the user is located must be determined to obtain the number of cells required to encompass the query range and produce the minimum obfuscation areas. First, three straight lines passing through a random cell in the cellular-structure map are conceptualized (the three red lines in the cell illustrated in Figure 3). The three lines divide the cell into six equilateral triangles. Without loss of generalizability, the linear equations of the three straight lines intersecting the cell containing the current location of the user (i, j) can be used to determine the equilateral triangle with the user (Figure 8):

Linear equation of $\overline{P^1P^4}$, $f_0(x, y)$: $x - H_x^{(i,j)} = 0$

Linear equation of $\overline{P^2P^5}$, $f_1(x, y)$: $(x - H_x^{(i,j)}) - \sqrt{3}(y - H_y^{(i,j)}) = 0$

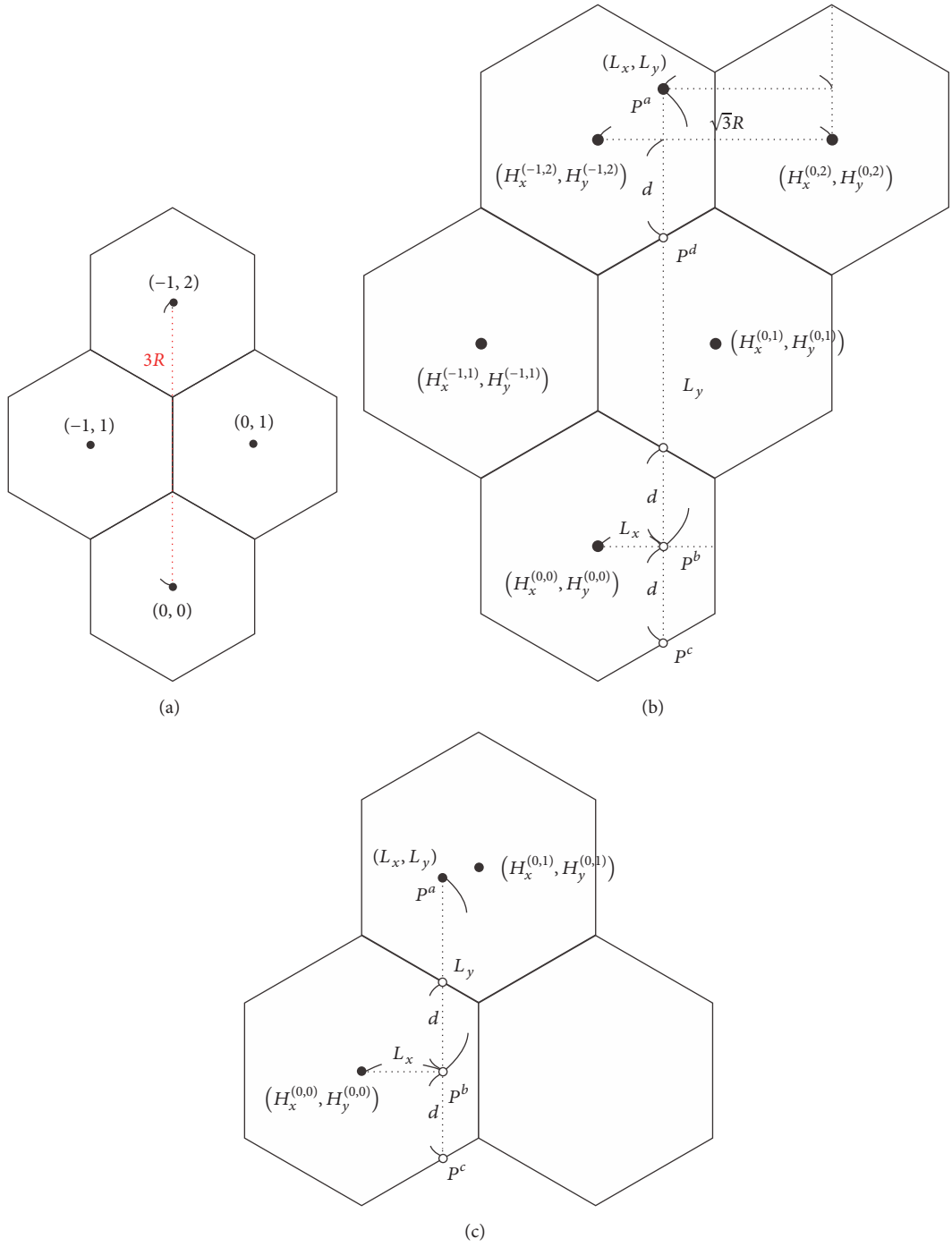


FIGURE 5: Calculating the distance between cells; (a) distance of vertically adjacent cells; (b) user within the same cell ($e = 0$); (c) user crossing over to another cell ($e = 1$).

Linear equation of $\overline{P^3P^6}$, $f_2(x, y)$: $(x - H_x^{(i,j)}) + \sqrt{3}(y - H_y^{(i,j)}) = 0$

For example, when $f_0(L_x, L_y) \geq 0$, $f_1(L_x, L_y) < 0$, and $f_2(L_x, L_y) > 0$, $f_0(L_x, L_y)$ represent that the user is either on or to the right of $\overline{P^1P^4}$; $f_1(L_x, L_y)$ indicates that the user is on $\overline{P^2P^5}$; and $f_2(L_x, L_y)$ means that the user is on $\overline{P^3P^6}$. A combined analysis of the three lines shows that the user is in $\Delta P^1P^2P^7$ of Tir = 1 (Figure 8(a)). Thus, four cells are selected as the obfuscation area for the user's query. These cells are numbered (i, j), $(i, j+1)$, $(i+1, j)$, and $(i-1, j+1)$. If the user's location is at the center of the cell ($L_x = H_x^{(i,j)}$) and $L_y = H_y^{(i,j)}$, a random equilateral triangle can be represented the user's query location. In Algorithm 1, QH_p^t represents the obfuscation area of query t in p, and $p = 1$ is the obfuscation area of the location provided by the user.

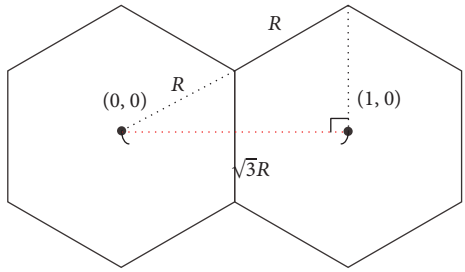


Figure 6: Distance between two horizontally adjacent cells.

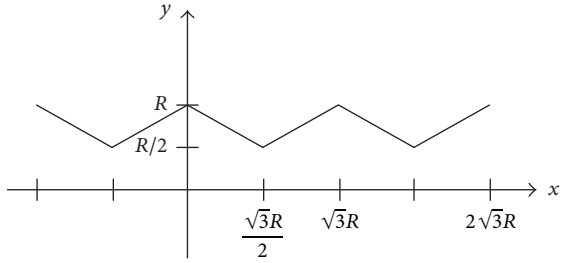


Figure 7: Effects of L_x on d.

Subsequently, whether the four cells encompass the user's query range must be determined. In Algorithm 1, three cells neighboring a random triangular section of (i, j) in Figure 8 are selected to form four cells. Because of the similarities among the three triangles, a random location in the upper-right triangular section of (i, j) is selected, without loss of generalizability, to verify that the combined area of the four shaded cells is the minimum obfuscation area to encompass the user's query range (Figure 9).

Supporting Theorem 1. *If the query range center is P^a, the radius of the query range is $(\sqrt{3}/2)R$. The user is in a random location in a triangular section with points $P^1 = (H_x^{(i,j)}, H_y^{(i,j)} + R)$, $P^2 = (H_x^{(i,j)} + (\sqrt{3}/2)R, H_y^{(i,j)} + (1/2)R)$, and $P^7 = (H_x^{(i,j)}, H_y^{(i,j)})$ and a side length R (Figure 9). Subsequently, the query range must include the cell with the triangular section and three neighboring cells to create an OA comprising four cells, specifically $(H_x^{(i,j)}, H_y^{(i,j)})$, $(H_x^{(i+1,j)}, H_y^{(i+1,j)})$, $(H_x^{(i-1,j+1)}, H_y^{(i-1,j+1)})$, and $(H_x^{(i,j+1)}, H_y^{(i,j+1)})$.*

Proof. In Figure 9, if P^a is located between $P^1 = (H_x^{(i,j)}, H_y^{(i,j)} + R)$ and $P^2 = (H_x^{(i,j)} + (\sqrt{3}/2)R, H_y^{(i,j)} + (1/2)R)$ on a line expressed as $x + \sqrt{3}y = \sqrt{3}R$, then a parallel line $(x + \sqrt{3}y = 2\sqrt{3}R)$ with a vertical distance of $(\sqrt{3}/2)R$ can be determined. Subsequently, two points can be observed on $x + \sqrt{3}y = 2\sqrt{3}R$, namely, $P^9 = (H_x^{(i,j)}, H_y^{(i,j)} + 2R)$ and $P^{11} = (H_x^{(i,j)} + \sqrt{3}R, H_y^{(i,j)} + R)$, which denotes that $|\overline{P^2P^{11}}| = |\overline{P^1P^9}| = R$ and the vertical distance from $\overline{P^1P^2}$ to $\overline{P^9P^{11}}$ is $(\sqrt{3}/2)R$. Similarly, a parallel line to $\overline{P^1P^7}$ with a distance of $(\sqrt{3}/2)R$ can be observed $(\overline{P^5P^8})$. Therefore, $|\overline{P^1P^8}| = |\overline{P^5P^7}| = R$. A parallel line to $\overline{P^2P^7}$ with a distance of $(\sqrt{3}/2)R$ can be observed $(\overline{P^4P^{11}})$. Therefore, $|\overline{P^4P^7}| = |\overline{P^2P^{12}}| = R$. Subsequently, the vertical distances from P^1 to $\overline{P^9P^{11}}$, from P^2 to $\overline{P^4P^{12}}$, and from P^7 to $\overline{P^5P^8}$ are all $(\sqrt{3}/2)R$. Thus, a circular query range with a radius of $(\sqrt{3}/2)R$ and a center point anywhere within the triangle created by P^1, P^2, and P^7 inevitably encompasses a section of the hexagonal section created by P^4, P^5, P^8, P^9, P^{11}, and P^{12}. Moreover, the polygon created by P^4, P^5, P^8, P^9, P^{11}, and P^{12} encompasses $(H_x^{(i,j)}, H_y^{(i,j)})$, $(H_x^{(i+1,j)}, H_y^{(i+1,j)})$, $(H_x^{(i-1,j+1)}, H_y^{(i-1,j+1)})$, and $(H_x^{(i,j+1)}, H_y^{(i,j+1)})$. Therefore, Supporting Theorem 1 holds. □

Supporting Theorem 1 confirms that the obfuscation areas generated using the four cells encompasses the user's query range. We subsequently developed an additional theorem to test whether a fewer number of cells can be used to encompass the user's query range.

Supporting Theorem 2. *If the query range center is P^a, the radius of the query range is $(\sqrt{3}/2)R$. The user is in a random location within a triangular section with points $P^1 = (H_x^{(i,j)}, H_y^{(i,j)} + R)$, $P^2 = (H_x^{(i,j)} + (\sqrt{3}/2)R, H_y^{(i,j)} + (1/2)R)$, and $P^7 = (H_x^{(i,j)}, H_y^{(i,j)})$ and a side length R (Figure 9). Subsequently, at least four cells are required to encompass the user's query range.*

Proof. Assume that P^a is in a random location in $\Delta P^1P^2P^7$. Subsequently, only three cells are required to encompass the user's query range. From Figure 9, the center point of the P^a on $\overline{P^1P^2}$ can be identified. With only three cells, the diameter of the query range must be less than or equivalent to R of $\overline{P^1P^2}$ (query diameter $\leq R$). In actuality, the query diameter is greater than R $(\sqrt{3}R > R)$. Thus, this hypothesis is contrary to fact, verifying that at least four cells are required for the obfuscation area to encompass the users query range. □

Theorem 1. *If the user appears in a random location on the map, his or her query range is a circle with a radius of $(\sqrt{3}/2)R$. The proposed algorithm can use the lowest number of cells to encompass the user's query range. The algorithm can maintain one-third of the size of the obfuscation area when the obfuscation algorithm is known to the attacker.*

Proof. Supporting Theorem 1 indicates that an obfuscation area comprising four cells could sufficiently encompass the user's query range when the user is located in a random location in $\Delta P^1P^2P^7$. Supporting Theorem 2 indicates that at least four obfuscated cells are required in order to sufficiently encompass the user's query range when the user is located in a random location in $\Delta P^1P^2P^7$. Naturally, the user must be in $\Delta P^1P^2P^7$ or $\Delta P^1P^2P^{10}$ for the LBS server to produce the shaded obfuscation areas with the algorithm (Figure 9). The combined area of the two triangles is guaranteed to be one-third of the obfuscation areas. □

The user is located within a cell comprising six equilateral triangles. Therefore, the location of the user in $\Delta P^1P^2P^7$ is verified regardless of which triangle the user is located in.

Obfuscation Area
Input: User position $P^a(L_x, L_y)$, User Cell No. (i_{P^a}, j_{P^a})
Output: QH
(1) $(i, j) = (i_{P^a}, j_{P^a})$
(2) Tir = 0
(3) **if** $(L_x = H_x^{(i,j)}$ **and** $L_y = H_y^{(i,j)})$
(4) Tir = Random $[1, 2, \ldots, 6]$
(5) **end if**
(6) **if** $((f_0(L_x, L_y) \geq 0$ **and** $f_1(L_x, L_y) < 0)$ **or** Tir = 1)
(7) QH = $\{(i, j), (i, j+1), (i+1, j), (i-1, j+1)\}$
(8) **else if** $((f_0(L_x, L_y) > 0$ **and** $f_2(L_x, L_y) > 0)$ **or** Tir = 2)
(9) QH = $\{(i, j), (i+1, j), (i, j+1), (i+1, j-1)\}$
(10) **else if** $((f_0(L_x, L_y) > 0$ **and** $f_1(L_x, L_y) > 0$ **and** $f_2(L_x, L_y) \leq 0)$ **or** Tir = 3)
(11) QH = $\{(i, j), (i+1, j), (i, j-1), (i+1, j-1)\}$
(12) **else if** $((f_0(L_x, L_y) \leq 0$ **and** $f_1(L_x, L_y) > 0$ **and** $f_2(L_x, L_y) < 0)$ **or** Tir = 4)
(13) QH = $\{(i, j), (i-1, j), (i, j-1), (i+1, j-1)\}$
(14) **else if** $((f_0(L_x, L_y) < 0$ **and** $f_1(L_x, L_y) \leq 0$ **and** $f_2(L_x, L_y) < 0)$ **or** Tir = 5)
(15) QH = $\{(i, j), (i-1, j), (i, j-1), (i-1, j+1)\}$
(16) **else**
(17) QH = $\{(i, j), (i-1, j), (i, j+1), (i-1, j+1)\}$
(18) **end if**
(19) **return** QH

Algorithm 1: Generating an obfuscation area to cover the query range of users.

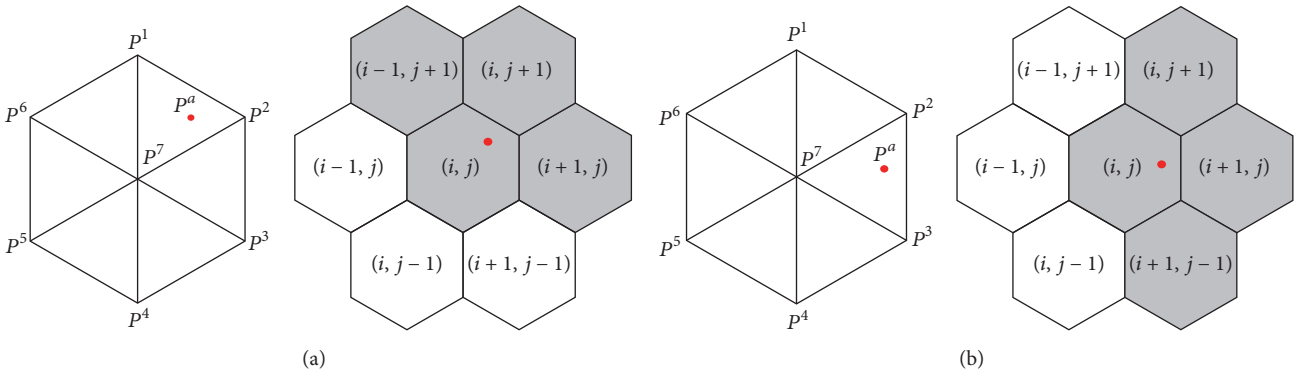


Figure 8: The obfuscation area: (a) user in Tir = 1; (b) user in Tir = 2.

This suggests that if the user is located anywhere on the map, the proposed obfuscation algorithm produces an obfuscation area of at least four cells, which is the lowest number of cells required, and guarantees that the area of the cells is at least one-third of the obfuscation areas.

3.3. Producing the Obfuscation Areas of Multiple Pseudoqueries. Section 3.2 describes how an obfuscated query area is produced to prevent attackers from obtaining the locations of users in sensitive areas such as special clinics or gyms. Users largely assume that attackers have a 1 in k chance of intercepting submitted queries (k-anonymity). Thus, we developed an algorithm that can produce multiple pseudoqueries to satisfy users' k-anonymity settings. To enhance the relevance of the pseudoqueries and reduce the number of obfuscation areas, we developed an algorithm that produces multiple pseudoqueries in batches so that individual obfuscation areas and queries can serve as pseudoqueries for other users. Finally, the algorithm replenishes inadequate queries while satisfying individual privacy requirements.

When the anonymizer receives t privacy requests $(k_1^t, k_2^t, \ldots, k_u^t)$ in T from u users in different locations $(\mathrm{CN}_1^t, \mathrm{CN}_2^t, \ldots, \mathrm{CN}_u^t)$ and no matches are cached, these queries must be transmitted to the LBS server. The anonymizer uses the proposed algorithm (Algorithm 1) to generate different obfuscation areas for the users $(\mathrm{QH}_1^t, \mathrm{QH}_2^t, \ldots, \mathrm{QH}_u^t)$. If the users collectively form an OA, then $\mathrm{QS}^t = \{\mathrm{QH}_1^t, \mathrm{QH}_2^t, \ldots, \mathrm{QH}_u^t\}$ can satisfy the maximum privacy requirement of the users.

$$k_{\mathrm{MAX}}^t = \max\left(k_1^t, k_2^t, \ldots, k_u^t\right). \tag{6}$$

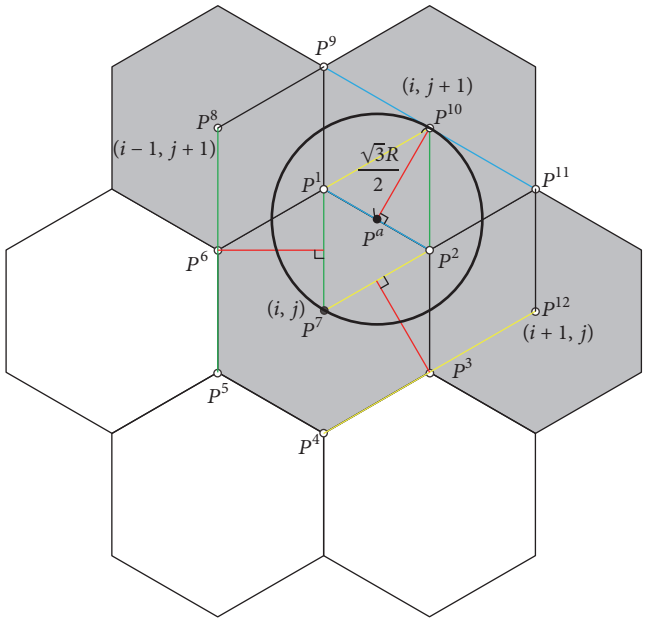


Figure 9: Query range and the obfuscation area.

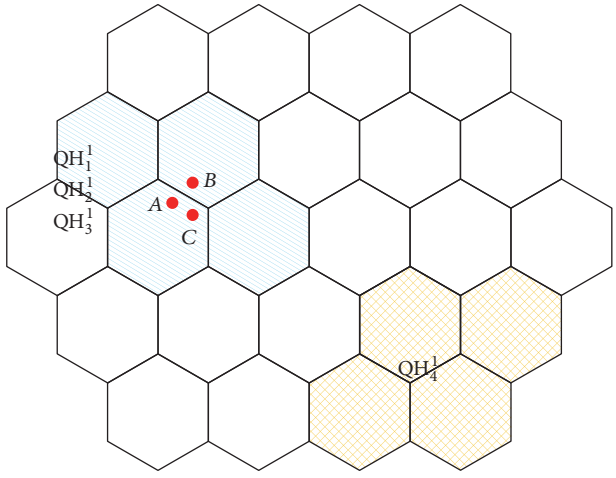


Figure 10: Overlapping user location causing inadequate privacy strength.

In other words, the OA collectively formed by the u users must contain four times as many cells than the number of cells required for the maximum privacy requirements.

$$\left|QH_1^t \cup QH_2^t \cup \cdots \cup QH_u^t\right| \geq 4 * k_{MAX}^t. \quad (7)$$

The privacy requirements of u users can be satisfied by combining the obfuscation areas of their query locations. However, the number of obfuscation areas must be generated when too few users are available or when users are close together (blue area in Figure 10). For example, Users A, B, and C in Figure 10 request a privacy strength of only 2. Therefore, $k_1^t = k_2^t = k_3^t = 2$. However, the three users area within the same obfuscation area generated by the anonymizer, causing $|QH_1^t \cup QH_2^t \cup QH_3^t| = 4$. In this instance, a pseudoobfuscation area consisting of four cells must be generated (Figure 10) to meet the obfuscated cell requirement of $4 * k_{MAX}^t = 8$.

To generate a pseudoobfuscation area that meets the privacy requirements, we developed a method for producing multiuser pseudoobfuscation areas (Algorithm 2). The method follows three criteria to repeatedly produce obfuscation areas until the obfuscation requirement of k_{MAX}^t is met:

(1) Avoid VBAs [3] in the center location of the pseudoobfuscation area generated for the user's location.

(2) Avoid generating pseudoobfuscation areas already cached in the anonymizer. Based on the open obfuscation area generation algorithm, attackers know that the queries transmitted to the LBS server are not cached in the anonymizer. Subsequently, the LBS server deduces the cache data of the anonymizer by using the open cache algorithm. Therefore, the pseudoqueries that are detected as cached queries by the LBS server are filtered out.

(3) Create new pseudocell numbers (CN_{dummy}^t) three layers from the user cell (CN_{sel}^t) to avoid generating an obfuscation area that overlaps CN_{dummy}^t and CN_{sel}^t and reinforce obfuscation strength more rapidly. Therefore, the anonymizer randomly selects one out of six cells three layers away from CN_{sel}^t, namely, $(i_{sel} + 2, j_{sel} + 1)$, $(i_{sel} + 3, j_{sel} - 2)$, $(i_{sel} + 1, j_{sel} - 3)$, $(i_{sel} - 2, j_{sel} - 1)$, $(i_{sel} - 3, j_{sel} + 2)$, and $(i_{sel} - 1, j_{sel} + 3)$, as CN_{dummy}^t to generate the obfuscation area.

A cell is randomly selected from CN^t to serve as the center point (Row (2)) to meet Criterion 1. Then, a pseudocell number CN_{dummy}^t (Row (3)) is randomly selected from the cells surrounding CN_{sel}^t to meet Criteria 2 and 3. Subsequently, CN_{dummy}^t must contain an intersection. The pseudocell number CN_{dummy}^t is added to CN^t so that $CN^t = CN^t + CN_{dummy}^t$ (Row (5)), and the pseudoobfuscation area generated using CN_{dummy}^t is added to QS^t (Rows (6) to (9)). This process is repeated until QH_q^t is generated to meet the obfuscation requirement of k_{MAX}^t. Finally, an obfuscation area set is produced.

$$QS^t = \left\{QH_1^t, QH_2^t, \ldots, QH_q^t \mid \forall (i, j) \in QH_a^t, \left|V_{(i,j)}\right| \neq 0, \ 1 \leq a \leq q\right\}. \quad (8)$$

In Figure 11, Users A, B, and C move along the red line and transmit queries at the red points at different times. The blue, yellow, and red areas represent the three obfuscation areas generated by the anonymizer for the users' queries transmitted to the LBS server. The anonymizer receives the privacy requirements of Users A and B, which are $k_A^1 = 1$ and $k_B^1 = 2$. However, QH_1^1 and QH_2^1 generated for the locations of Users A and B overlap, creating an OA with only seven cells. This fails to meet User B's privacy requirement of 2, which requires eight cells ($4 * k_B^1 = 8$). The anonymizer selects a random cell ($CN_B^t = (1, 1)$) three layers away from CN_A^t and

Generate k^t_{MAX} Obfuscation Area
Input: k^t_{MAX}, $\text{CN}^t = \{\text{CN}^t_1, \text{CN}^t_2, \ldots, \text{CN}^t_u\}$,
$\text{QS}^t = \{\text{QH}^t_1, \text{QH}^t_2, \ldots, \text{QH}^t_u\}$,
$\text{NumOfCells} = |\text{QH}^t_1 \cup \text{QH}^t_2 \cup \cdots \cup \text{QH}^t_u|$
Output: QS^t
(1) **while** $(\text{NumOfCells} < 4 * k^t_{\text{MAX}})$
(2) $\quad \text{CN}^t_{\text{sel}} = \text{Random}(\text{CN}^t)$
(3) $\quad \text{CN}^t_{\text{dummy}} = \text{FindDummyCell}(\text{CN}^t_{\text{sel}})$
(4) $\quad$ **if** $(\text{CN}^t_{\text{dummy}} \neq \text{null})$
(5) $\quad\quad \text{CN}^t = \text{CN}^t + \text{CN}^t_{\text{dummy}}$
(6) $\quad\quad (d_x, d_y) = \text{Random}(V_{\text{CN}^t_{\text{dummy}}})$
(7) $\quad\quad \text{temp QS}^t = \text{Obfuscation Area}((d_x, d_y), \text{CN}^t_{\text{dummy}})$
(8) $\quad\quad$ **if** $(\text{temp QS}^t \notin \text{Cache})$
(9) $\quad\quad\quad \text{QS}^t = \text{QS}^t + \text{temp QS}^t$
(10) $\quad\quad\quad \text{NumOfCells} = |\text{QS}^t|$
(11) $\quad\quad$ **end if**
(12) $\quad$ **end if**
(13) **end while**
(14) **return** QS^t

FindDummyCell
Input: $(\text{CN}^t_{\text{sel}})$
(1) $\text{side} = \text{Random}[1, 2, \ldots, 6]$
(2) **if** $(\text{side} = 1$ **and** $V_{(i_{\text{sel}}+2, j_{\text{sel}}+1)} \neq \emptyset)$
(3) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} + 2, j_{\text{sel}} + 1)$
(4) **else if** $(\text{side} = 2$ **and** $V_{(i_{\text{sel}}+3, j_{\text{sel}}-2)} \neq \emptyset)$
(5) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} + 3, j_{\text{sel}} - 2)$
(6) **else if** $(\text{side} = 3$ **and** $V_{(i_{\text{sel}}+1, j_{\text{sel}}-3)} \neq \emptyset)$
(7) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} + 1, j_{\text{sel}} - 3)$
(8) **else if** $(\text{side} = 4$ **and** $V_{(i_{\text{sel}}-2, j_{\text{sel}}+1)} \neq \emptyset)$
(9) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} - 2, j_{\text{sel}} - 1)$
(10) **else if** $(\text{side} = 5$ **and** $V_{(i_{\text{sel}}-3, j_{\text{sel}}+2)} \neq \emptyset)$
(11) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} - 3, j_{\text{sel}} + 2)$
(12) **else if** $(\text{side} = 6$ **and** $V_{(i_{\text{sel}}-1, j_{\text{sel}}+3)} \neq \emptyset)$
(13) $\quad \text{CN}^t_{\text{dummy}} = (i_{\text{sel}} - 1, j_{\text{sel}} + 3)$
(14) **else**
(15) $\quad \text{CN}^t_{\text{dummy}} = \text{null}$
(16) **end if**
(17) **return** $\text{CN}^t_{\text{dummy}}$

Algorithm 2: Producing an OA that satisfies all users.

CN^t_B. It identifies $(i_B - 2, j_B - 1) = (-1, 0)$ and uses this cell to generate a pseudoobfuscation area (QH^1_3). This generates an obfuscation area with $11 > 4 * k^1_B$ (blue area in Figure 10), which meets the privacy requirements of Users A and B.

Then, the anonymizer separately receives the privacy requirements of $k^2_A = 5$, $k^2_B = 2$, and $k^2_C = 3$ from Users A, B, and C, respectively. Because the query of User A is already cached in the anonymizer, it can directly respond to that query. It then generates obfuscation areas for Users B and C and transmits them to the LBS server. Therefore, three obfuscation areas are created to satisfy the requirement of $4 * k^2_C = 12$ obfuscated cells, namely, QH^2_1, QH^2_2, and QH^2_3 (yellow area in Figure 11).

Finally, the anonymizer receives the privacy requirements of $k^2_A = 1$ and $k^2_B = 2$ from Users A and B, respectively. Because the query of User B is already cached in the anonymizer, it generates an obfuscation area only for User A in order to satisfy the two obfuscation requirements (red area in Figure 11).

The preceding obfuscation method have two problems. First, the anonymizer can immediately respond to the user without accessing the LBS server when a similar query is cached. Existing methods aimed at enhancing the cache hit ratio [25–27] effectively reduce the likelihood of exposing queries to the LBS server while conserving the communication cost and computation load of the anonymizer. For example, the proposed method uses a hierarchical clustering method [28–31] to group the cached queries according to popularity. These groups are then used to generate corresponding pseudoqueries to prevent attacks that exploit an uneven query distribution [32].

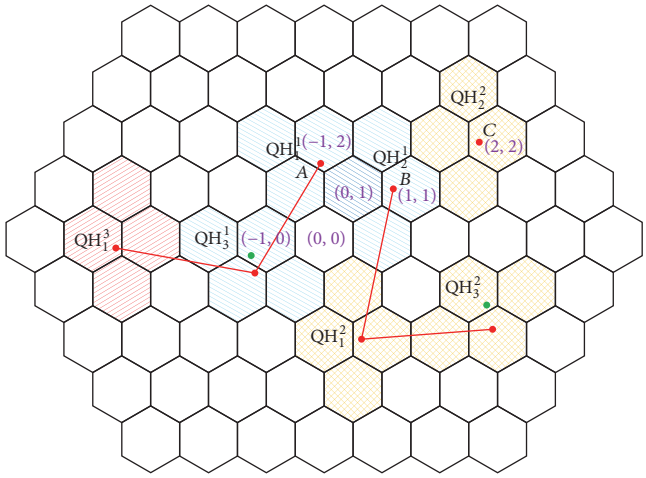


FIGURE 11: Unchained obfuscation area generated for the continuous queries of three users.

Second, when the anonymizer transmits user IDs to the LBS server, attackers can determine users' travel routes by analyzing the queries of similar IDs, even when the location of the user is obfuscated. The following section proposes a method for generating unrepeated random pseudouser IDs for each QH_q^t.

3.4. Generating Obfuscated Query Information. We developed a method to prevent LBS servers from combining obfuscation areas and user IDs to deduce users' travel routes. Even when a simple algorithm is applied to substitute different user IDs with the same ID, LBS servers can still combine intersection and traffic speed-limit data to deduce users' travel range and travel routes [33–36]. To prevent this problem, when the anonymizer generates q obfuscation areas for u^h user queries ($QS^t = \{QH_1^t, QH_2^t, \ldots, QH_q^t\}$) to satisfy their user privacy requirements, it randomly produces the q pseudo-IDs:

$$\begin{aligned} VID^t = \left\{VID_1^t, VID_2^t, \ldots, VID_q^t \mid q \in \mathbf{N}^+\right\}, \\ \forall VID_y^t, VID_z^t \in VID^t,\ VID_y^t \neq VID_z^t. \end{aligned} \tag{9}$$

Then, the anonymizer combines all obfuscation areas and the corresponding POIs to generate

$$\begin{aligned} &LocPoi^t \\ &= \left\{QH_1^t \parallel POI_1^t, QH_2^t \parallel POI_2^t, \ldots, QH_q^t \parallel POI_q^t\right\}. \end{aligned} \tag{10}$$

The content is randomly interchanged to generate

$$\begin{aligned} &LocPoi^{t\prime} \\ &= \left\{QH_1^{t\prime} \parallel POI_1^{t\prime}, QH_2^{t\prime} \parallel POI_2^{t\prime}, \ldots, QH_q^{t\prime} \parallel POI_q^{t\prime}\right\}. \end{aligned} \tag{11}$$

Directly transmitting the query without changing the order of $LocPoi^t$ allows the LBS server to use the known algorithm to identify QH_1^t to be the real user location. Changes are logged with the anonymizer and used to filter user query results once they are returned by the LBS server. Finally, VID^t and $LocPoi^{t\prime}$ are combined to transmit the protected query to the LBS server:

$$Query^t = \left\{VID^t, LocPoi^{t\prime}\right\}. \tag{12}$$

4. Analysis

This section analyzes the security and performance of the proposed method and compares the results with those of previous studies. In Section 4.1, we present the security analysis items and compare past security problems. In Section 4.2, the method is applied to a map to examine the method's real-time performance.

4.1. Security Analysis. The unchained location privacy protection method developed in the present study was based on a trusted anonymizer and existing user/anonymizer security architectures to protect information confidentiality. Therefore, this section discusses four threat models derived from attacks that occur during the communication between the trusted LBS server and anonymizer. The results verify that the proposed method can effectively guard against most LBS attacks when the algorithm is known to the attacker.

When attackers possess the background knowledge of the maps and the capacity to continuously monitor user query content, they can issue the following attacks on user privacy:

Location Homogeneity Attack (LHA). Attackers collect queries from a particularly sensitive area to collect user information, such as a hospital specializing in cardiology and heart surgery, to gain information on heart patients.

Map Matching (MM). Attackers use background knowledge to filter out unlikely query source locations (e.g., lakes) to enhance the likelihood of identifying the actual locations of users.

When LBS servers and general attackers use known location obfuscation algorithms to analyze the queries submitted by multiple users in obfuscated locations, they can perform the following attacks on user privacy:

Known Algorithm Attack (KAA). Attackers who are aware of the obfuscation algorithm can use the algorithm to calculate the obfuscation areas generated in different locations and filter out the less likely results to reduce the obfuscation strength of user locations.

Distance VBA. Attackers calculate the center points of obfuscation areas to estimate the actual locations of users [3].

When LBS servers and general attackers cross-reference the obfuscation areas of queries submitted by different IDs in different locations at different times, they can perform the following attacks on user privacy.

Maximum Movement Boundary (MMB). Attackers examine the traffic speed limits of the map to calculate the maximum movement boundary of the user. They eliminate the areas

Table 1: Security comparison chart for multi-LBS queries.

Protocols	Attacker knowledge					
	LHA	MM	MQA	MMB	VBA	KA
Xu and Cai [21]	X	O	O	O	X	X
Xu and Cai [14]	O	O	O	O	X	X
Shankar et al. [15]	O	O	O	O	O	O
Wang and Liu [22]	O	O	X	O	O	X
Ardagna et al. [20]	O	O	O	X	O	O
Lee et al. [23]	O	O	O	O	X	X
Song et al. [24]	O	O	O	O	X	X
Niu et al. [25]	O	O	O	O	X	X
Niu et al. [3]	X	O	X	O	O	O
Our scheme	O	O	O	O	O	O

that the user cannot reach to reduce the obfuscation areas of continuous queries.

Multiple Query Attack (MQA). Attackers cross-reference the members and movement of users in different obfuscation areas to filter out pseudousers and identify real users.

The results in Table 1 show that the proposed method effectively guards against all known attacks. The symbol "O" denotes that the method can defend against this type of attack, and the symbol "X" denotes that the method fails to defend against this type of attack. In [3, 21], methods were proposed to obfuscate the locations of numerous querying users. However, these methods failed to consider user locations that approximate sensitive areas, which enables attackers to exploit these areas by using LHAs to obtain user locations. In [3, 22], algorithms were developed to obfuscate multiquery submissions. However, these methods could not continuously obfuscate locations when the user is moving, which enables attackers to observe the route of the users by performing MQAs. In [20], a method was proposed to substitute sensitive query locations with nearby insensitive locations cached in the anonymizer. However, this method failed to consider user movement speeds, enabling attackers to filter user locations by performing MMBs. Moreover, [20] used the center location of users to generate obfuscation areas, enabling attackers to estimate the actual location of users by performing VBAs [14, 21, 23–25]. Attackers could also confirm the center location of users in an obfuscation area once the algorithm is known to the attacker. In [22], a method was proposed for generating road network obfuscation areas by searching neighboring intersections to avoid placing users on the same road. However, systematically searching neighboring intersections enables attackers to perform KAs to map the obfuscation method and identify user locations.

4.2. Performance Analysis. We implemented simulations in Java 8 on a computer equipped with an Intel i5-4570 CPU to create a test environment with a road map of Oldenburg, Germany [37]. Figure 12 shows that the anonymizer expanded the side length of the map from 10 to 40 km while generating cells to satisfy R. Notably, reducing the R-value increased the

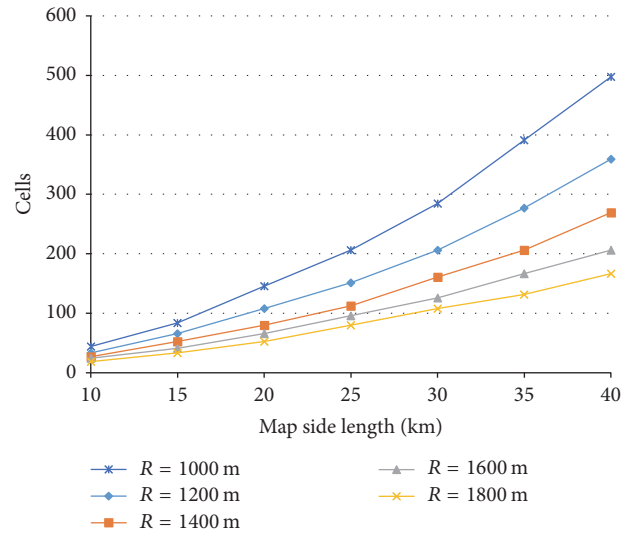


Figure 12: Effects of map size and cell size on the cell quantity.

number of cells generated on maps with similar side lengths, reducing the content of each cell. The proposed method uses the same number of cells to obfuscate user query range. Therefore, lower R-values reduce the user query range and decrease the amount of data required to return query results from the LBS server.

We observed intersection conditions by dividing the Oldenburg map into R-sized cells (Figure 13). Results showed that smaller cells contained fewer intersections. Although Figure 12 shows that cells with shorter sides reduce the transmission load, the results in Figure 13 indicate that smaller cells reduce the number of intersections per cell. Fewer intersections increase the likelihood of attackers estimating the actual location of users. Therefore, a balance between transmission efficiency and the privacy strength must be achieved.

In Figure 14, the privacy requirement of each user is assumed to be $k = 5$ and $R = 1200$ to compare the required average number of queries transmitted to the LBS server. Compared with the result of [25] regarding the number of queries submitted by a single user to generate an obfuscation

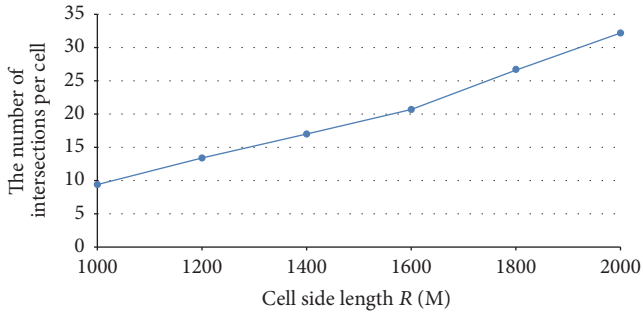


FIGURE 13: Effects of the Oldenburg map and cell side length (R) on the number of intersections per cell.

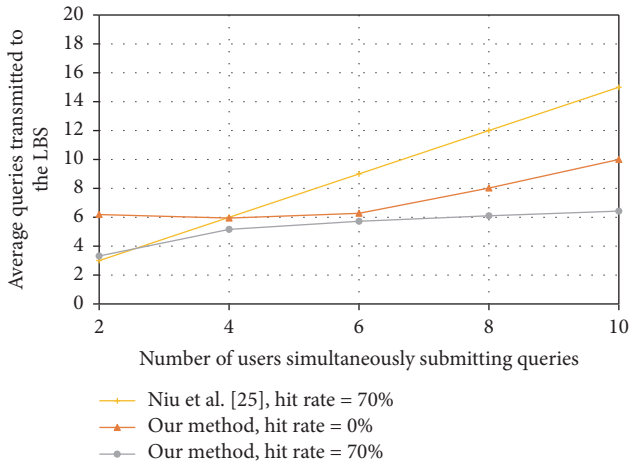


FIGURE 14: Comparing the number of users and the query volume transmitted to the LBS Server.

area, our cache hit ratio was 0, indicating that, without using the cache, four users or more are required to simultaneously transmit a query to meet the privacy requirements with a reduced number of pseudoqueries sent by the anonymizer to the LBS server. The proposed method can combine the user queries of similar obfuscation areas to meet various privacy requirements. In [25], a cache was used to reduce computation and transmission loads. In the present study, we adopted a cache hit ratio of 70%, similar to that used in [25]. Regardless of the number of users, we maintained the privacy protection strength equivalent to that reported in [25], and the performance of the proposed method improved as the number of users was increased. In our proposed method, the number of obfuscation areas must be generated when too few users are available or when users are close together. When the number of users is 2, only 3 queries are submitted to the LBS in [25], and our method requires 6.185 queries with hit rate = 0% or 3.32 queries with hit rate = 70%. But in our method, the obfuscation areas of the query locations can be combined when the number of users increases, which reduce the number of queries that needs to be sent to the LBS. In Figure 14, when the number of users = 8, 12 queries is submitted to the LBS in [25], our method needs 8.019 queries with hit rate = 0% and only 6.427 queries with hit rate = 70%. In this situation, our performance is better than [25].

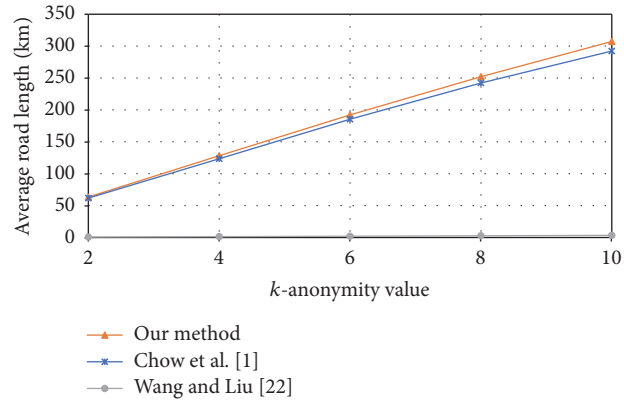


FIGURE 15: The relationship of the k-anonymity value and the average road length in the map.

Figure 15 shows that the average road lengths in the Oldenburg map that satisfy the k-anonymity when the radius $R = 1,200$. In [22], the roads were simply extended to obfuscate the location of users. Niu et al.'s method [3] uses a random walk-based cloaking algorithm, and the method proposed in the present study divides the map into cells. Therefore, the average road lengths of the overall obfuscated areas using the proposed method and [3] were markedly longer than that determined using the method proposed in [22]. Moreover, we generate extra queries to simulate a multiuser environment which requires generating additional obfuscation areas when the cells overlap. The proposed method generated 4.88% longer road length than [3] when $k = 10$.

5. Conclusion

We developed a privacy protection scheme to protect the real location suitable for moving users. The scheme produces multiuser pseudoqueries and uses obfuscation areas to prevent LBS servers from directly deducing users' real queries and precise locations. We verified that the method produces obfuscation areas with the least number of cells and guarantees one-third the original obfuscation areas size when the algorithm is disclosed. We also considered the distinct characteristic of user queries in different areas and adopted a grouping approach coupled with actual maps to reduce the likelihood of the pseudodata being filtered out by the LBS server, thereby satisfying users' privacy requirements. Furthermore, we incorporated a caching system to store users' continuous queries. The cache system coupled with multiuser queries prevents the LBS server from completing deducing users' routes. Instead, the LBS server can generate only scattered and obfuscated user locations. Therefore, the proposed method effectively protects location privacy during continuous querying. The cache approach also reduces the likelihood of user locations being transmitted to the LBS server, decreases the computation and transmission loads of the anonymizer, and enhances system performance. The proposed method is fully compatible with various user devices. They can use their original mobile devices and Internet

service providers to access the trusted anonymizer to protect their location details when submitting a query. Finally, we verified that the proposed method effectively protects users' identities, locations, and interests and guards against most currently known attacks on location privacy. We also used a real-time road map to test the proposed method. Figure 14 shows that the proposed method uses a cache approach to greatly reduce the amount of query information exposed to the LBS server. A summary of the results illustrated in Figures 14 and 15 shows that the proposed method outperformed other existing methods.

Notations

R: Cell side length
u: Number of real users
q: Index value, $q \in \mathbf{N}^+$
k_q^t: k-anonymity requirement of the user q in query t
k_{MAX}^t: k-anonymity requirement for the multiuser query t $k_{\mathrm{MAX}}^t = \max(k_1^t, k_2^t, \ldots, k_u^t)$
ID: User ID
VID_q: Pseudo-ID q randomly generated by the anonymizer
(i, j): Cell number
(i_q, j_q): Cell number q
$(H_x^{(i,j)}, H_y^{(i,j)})$: X- and Y-coordinates of cell (i, j)
$P^a(L_x, L_y)$: User's real current location
(L_x^u, L_x^u): The real location of the user u in a multiuser query
CN_q^t: Cell number of user q in a query t $\mathrm{CN}_q^t = (i_q, j_q)$
CN^t: Cell number set during a query t $\mathrm{CN}^t = \{\mathrm{CN}_1^t, \mathrm{CN}_2^t, \ldots, \mathrm{CN}_q^t\}$
Tir: Number of the equilateral triangles formed by the six vertices of the cell $\{\mathrm{Tir} \in \mathbf{N}^+ \mid 1 \le \mathrm{Tir} \le 6\}$
$V_{(i,j)}^{\mathrm{Tir}}$: Intersection set contained in triangle Tir in cell (i, j)
V: The set of all intersections $\{V_{(i,j)} \subseteq V \mid V_{(i,j)} = V_{(i,j)}^1 \cup V_{(i,j)}^2 \cup \cdots \cup V_{(i,j)}^6, |V_{(i,j)}| \neq \emptyset\}$
$E_{(i,j)}^{\mathrm{Tir}}$: Road section set of triangle Tir in cell (i, j)
E: All road section sets $\{E_{(i,j)} \subseteq E \mid E_{(i,j)} = E_{(i,j)}^1 \cup E_{(i,j)}^2 \cup \cdots \cup E_{(i,j)}^6, |E_{(i,j)}| \neq \emptyset\}$
d: Vertical distance between L_y and the lower or upper boundary of the cell
t: Current query in a continuous query, $t \in \mathbf{N}^+$
T: Wait time of the anonymizer before receiving a query from a user
e: Indicator of additional space between cells, $e = \{0, 1\}$
QH_q^t: One obfuscation area comprising four cells, all cell numbers within the obfuscation area in q of query t
QS^t: OA set of query t $\mathrm{QS}^t = \{\mathrm{QH}_1^t, \mathrm{QH}_2^t, \ldots, \mathrm{QH}_q^t \mid \forall (i, j) \in \mathrm{QH}_a^t, |V_{(i,j)}| \neq 0, 1 \le a \le q\}$
LocPoi^t: QH_q^t and POI_1^t sets (no changes to production order) $\mathrm{LocPoi}^t = \{\mathrm{QH}_1^t \parallel \mathrm{POI}_1^t, \mathrm{QH}_2^t \parallel \mathrm{POI}_2^t, \ldots, \mathrm{QH}_q^t \parallel \mathrm{POI}_q^t\}$
$\mathrm{LocPoi}^{t\prime}$: Content order after changing LocPoi^t $\mathrm{LocPoi}^{t\prime} = \{\mathrm{QH}_1^{t\prime} \parallel \mathrm{POI}_1^{t\prime}, \mathrm{QH}_2^{t\prime} \parallel \mathrm{POI}_2^{t\prime}, \ldots, \mathrm{QH}_q^{t\prime} \parallel \mathrm{POI}_q^{t\prime}\}$.

Conflicts of Interest

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

Acknowledgments

This research was supported by the National Science Council of Taiwan under Grants nos. MOST 106-2221-E-130-001, MOST 106-3114-E-011-003, and MOST 106-2221-E-033-002.

References

[1] C.-Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems (ACM-GIS '06)*, pp. 171–178, ACM, November 2006.

[2] C.-Y. Chow, M. F. Mokbel, and X. Liu, "Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments," *GeoInformatica*, vol. 15, no. 2, pp. 351–380, 2011.

[3] B. Niu, X. Zhu, Q. Li, J. Chen, and H. Li, "A novel attack to spatial cloaking schemes in location-based services," *Future Generation Computer Systems*, vol. 49, pp. 125–132, 2015.

[4] A. Pfitzmann and M. Köhntopp, "Anonymity, unobservability, and pseudonymity—a proposal for terminology," in *Proceedings of International Workshop on Design Issues in Anonymity and Unobservability Berkeley*, vol. 2009, pp. 1–9, Springer, Berlin, Germany.

[5] T. Rodden, A. Friday, H. Muller, and A. Dix, "A lightweight approach to managing privacy in location-based services," Technical Report Equator-02-058, University of Nottingham, Lancaster University, University of Bristol, 2002.

[6] C. A. Ardagna, M. Cremonini, S. De Capitani Di Vimercati, and P. Samarati, "An obfuscation-based approach for protecting location privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 1, pp. 13–27, 2011.

[7] M. L. Damiani, E. Bertino, and C. Silvestri, "Protecting location privacy against spatial inferences: the PROBE approach," in *in Proceedings of the 2nd SIGSPATIAL ACM GIS 2009 International Workshop on Security and Privacy in GIS and LBS*, pp. 32–41, 2009.

[8] M. Duckham and L. Kulik, "A formal model of obfuscation and negotiation for location privacy," in *Proceedings of International Conference of Pervasive Computing*, pp. 152–170, May 2005.

[9] D. C. Howe and H. Nissenbaum, "TrackMeNot: resisting surveillance in web search," in *Lessons from the Identity Trail: Anonymity, Privacy, and Identity in a Networked Society*, pp. 417–436, 2009.

[10] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, "Preventing location-based identity inference in anonymous spatial queries," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 12, pp. 1719–1733, 2007.

[11] J.-H. Um, H.-D. Kim, and J.-W. Chang, "An advanced cloaking algorithm using Hilbert curves for anonymous location based service," in *Proceedings of the 2nd International Conference on Social Computing*, pp. 1093–1098, Minneapolis, MN, USA, August 2010.

[12] C. Zhang and Y. Huang, "Cloaking locations for anonymous location based services: a hybrid approach," *GeoInformatica*, vol. 13, no. 2, pp. 159–182, 2009.

[13] C.-P. Wu, C.-C. Huang, J.-L. Huang, and C.-L. Hu, "On preserving location privacy in mobile environments," in *Proceedings of the 2011 9th IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2011*, pp. 490–495, Seattle, WA, USA, March 2011.

[14] T. Xu and Y. Cai, "Exploring historical location data for anonymity preservation in location-based services," in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM '08)*, pp. 547–555, IEEE, April 2008.

[15] P. Shankar, V. Ganapathy, and L. Iftode, "Privately querying location-based services with sybilquery," in *Proceedings of the 11th ACM International Conference on Ubiquitous Computing, UbiComp'09*, pp. 31–40, usa, October 2009.

[16] B. Palanisamy and L. Liu, "MobiMix: protecting location privacy with mix-zones over road networks," in *Proceedings of the IEEE 27th International Conference on Data Engineering*, pp. 494–505, Hannover, Germany, April 2011.

[17] K.-T. Yang, G.-M. Chiu, H.-J. Lyu, D.-J. Huang, and W.-C. Teng, "Path privacy protection in continuous location-based services over road networks," in *Proceedings of the IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '12)*, pp. 435–442, October 2012.

[18] T.-H. You, W.-C. Peng, and W.-C. Lee, "Protecting moving trajectories with dummies," in *Proceedings of the 8th International Conference on Mobile Data Management (MDM '07)*, pp. 278–282, Mannheim, Germany, May 2007.

[19] A. Pingley, N. Zhang, and X. Fu, "Protection of query privacy for continuous location based services," in *Proceedings of the INFOCOM*, pp. 1710–1718, IEEE, Shanghai, China, 2011.

[20] C. Ardagna, G. Livraga, and P. Samarati, "Protecting privacy of user information in continuous location-based services," in *Proceedings of the IEEE 15th International Conference on Computational Science and Engineering (CSE '12)*, pp. 162–169, Nicosia, Cyprus, December 2012.

[21] T. Xu and Y. Cai, "Location anonymity in continuous location-based services," in *Proceedings of the 15th ACM International Symposium on Advances in Geographic Information Systems (GIS '07)*, pp. 300–307, November 2007.

[22] T. Wang and L. Liu, "Privacy-aware mobile services over road networks," in *in Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 1042–1053, 2009.

[23] H. Lee, B.-S. Oh, H.-I. Kim, and J. Chang, "Grid-based cloaking area creation scheme supporting continuous location-based services," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC '12)*, pp. 537–543, March 2012.

[24] D. Song, J. Sim, K. Park, and M. Song, "A privacy-preserving continuous location monitoring system for location-based services," *International Journal of Distributed Sensor Networks*, vol. 11, no. 8, pp. 1–10, 2015.

[25] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Enhancing privacy through caching in location-based services," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications (IEEE INFOCOM '15)*, pp. 1017–1025, IEEE, May 2015.

[26] S. Amini, J. Lindqvist, J. Hong, J. Lin, E. Toch, and N. Sadeh, "Caché: caching location-enhanced content to improve user privacy," in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, pp. 197–210, ACM, 2011.

[27] X. Zhu, H. Chi, B. Niu, W. Zhang, Z. Li, and H. Li, "MobiCache: When k-anonymity meets cache," in *Proceedings of the 2013 IEEE Global Communications Conference, GLOBECOM 2013*, pp. 820–825, IEEE, Atlanta, GA, USA, December 2013.

[28] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping Multidimensional Data*, J. Kogan, C. Nicholas, and M. Teboulle, Eds., pp. 25–71, Springer, Berlin, Germany, 2006.

[29] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.

[30] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[31] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: hierarchical clustering using dynamic modeling," *Computer*, vol. 32, no. 8, pp. 68–75, 1999.

[32] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Proceedings of the IEEE Symposium on Security and Privacy, SP 2011*, pp. 247–262, Berkeley, Calif, USA, 2011.

[33] C.-Y. Chow and M. F. Mokbel, "Trajectory privacy in location-based services and data publication," *ACM SIGKDD Explorations Newsletter*, vol. 13, no. 1, pp. 19–29, 2011.

[34] E. Kaplan, T. B. Pedersen, E. Sava, and Y. Saygin, "Discovering private trajectories using background information," *Data and Knowledge Engineering*, vol. 69, no. 7, pp. 723–736, 2010.

[35] T. N. Phan, T. K. Dang, and J. Küng, "User privacy protection from trajectory perspective in location-based applications," in *Proceedings of Interdisciplinary Information Management Talks*, pp. 281–288, 2011.

[36] M. Wernke, P. Skvortsov, F. Dürr, and K. Rothermel, "A classification of location privacy attacks and approaches," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 163–175, 2014.

[37] T. Brinkhoff, "Oldenburg: nodes & edges," 2017, http://iapg.jade-hs.de/personen/brinkhoff/generator.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 9394613, 22 pages
https://doi.org/10.1155/2017/9394613

Research Article

Fault Activity Aware Service Delivery in Wireless Sensor Networks for Smart Cities

Xiaomei Zhang,[1,2] Xiaolei Dong,[3] Jie Wu,[4] Zhenfu Cao,[3] and Chen Lyu[5]

[1]*Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China*
[2]*College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China*
[3]*Shanghai Key Laboratory for Trustworthy Computing, East China Normal University, Shanghai, China*
[4]*Department of Computer and Information Sciences, Temple University, Philadelphia, PA, USA*
[5]*School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai, China*

Correspondence should be addressed to Xiaolei Dong; dongxiaolei@sei.ecnu.edu.cn

Received 10 April 2017; Revised 1 July 2017; Accepted 24 July 2017; Published 20 September 2017

Academic Editor: Damianos Gavalas

Wireless sensor networks (WSNs) are increasingly used in smart cities which involve multiple city services having quality of service (QoS) requirements. When misbehaving devices exist, the performance of current delivery protocols degrades significantly. Nonetheless, the majority of existing schemes either ignore the faulty behaviors' variability and time-variance in city environments or focus on homogeneous traffic for traditional data services (simple text messages) rather than city services (health care units, traffic monitors, and video surveillance). We consider the problem of fault-aware multiservice delivery, in which the network performs secure routing and rate control in terms of fault activity dynamic metric. To this end, we first design a distributed framework to estimate the fault activity information based on the effects of nondeterministic faulty behaviors and to incorporate these estimates into the service delivery. Then we present a fault activity geographic opportunistic routing (FAGOR) algorithm addressing a wide range of misbehaviors. We develop a leaky-hop model and design a fault activity rate-control algorithm for heterogeneous traffic to allocate resources, while guaranteeing utility fairness among multiple city services. Finally, we demonstrate the significant performance of our scheme in routing performance, effective utility, and utility fairness in the presence of misbehaving sensors through extensive simulations.

1. Introduction

Wireless sensor networks (WSNs) have been integrated with smart cities and play an important role in smart city by providing versatile applications through sensors. With the demands for living and security standard of a city, it has become necessary for WSNs to support a series of city services, such as health monitoring, electricity consumption, intelligent transportation, visual target tracking, and multicamera surveillance [1, 2]. Sensors that are randomly distributed in a network cooperate with each other to deliver service data via multihop routing and rate control to the sink, which can communicate with conventional networks, for instance, the Internet.

Built upon open wireless medium, multiple city services in WSNs are particularly vulnerable to attackers which are attracted by sensitive information, less infrastructure, privacy, and so forth. Many service delivery protocols have been proposed and evaluated for countering different types of misbehaving nodes [3, 4]; however, most studies largely ignored the uncertainties and variabilities in the city environment. It is not an easy job to characterize the dynamics of dynamic ongoing or unknown attacks in an intuitionist way. Moreover, recent works in [5, 6] have demonstrated that the attackers with fixed strategy cannot disguise themselves as members of a city and are then marked as the adversaries. Inconsistent behaviors may exist in an intelligent misbehaving sensor or adapt its strategy under random attacks in smart grids [7], stealthy attacks in WSN-based IoT [8], and dynamic ongoing attacks in smart cities [9]. Hence, the impact of misbehaving sensors is probabilistic and time-varying in many cases.

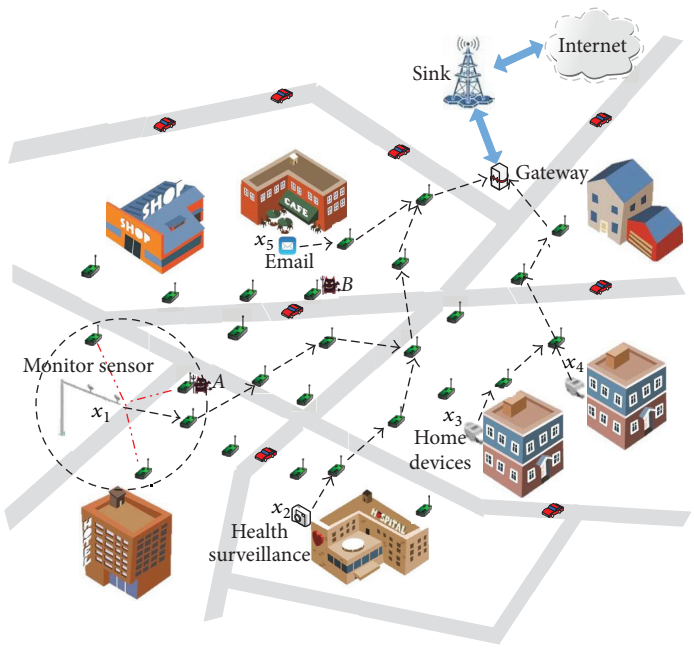


Figure 1: Multiservice delivery in a WSN of smart cities.

In order to characterize the effect of faulty behaviors on routing and throughput, we propose an impact collecting-based approach, which formulates the dynamics of faulty behaviors. A popular approach is to collect information about the direct impact of the misbehaviors, such as energy and delivery quality inside a sensor. Besides that, the delivery for city services is affected by some indirect impacts. For example, the vehicle misleads network routine and causes bandwidth consumption by announcing its various fake position simultaneously or the frequent time interval [10]. To defend against this type of misbehavior, a sensor needs to obtain trust verification from other sensors. The aim of our method is first to identify the state of a faulty sensor by, on direct impact and on indirect impact, gathering verification information received from its neighboring nodes. Then we model the state of being faulty at each sensor as a random process. Since the effect of faulty behaviors is probabilistic, the state of being faulty will also be nondeterministic and must be studied by applying a stochastic framework. Accordingly, we make each sensor establish novel metrics fault activity (FA) for modeling the stochastic state of being faulty in terms of statistical information about the probabilistic faulty nodes, which is also utilized to select next forwarding candidates for each hop and to allocate resource for each service.

Geographic opportunistic routing (GOR) is considered an effective and flexible way to improve network performance with the help of WSN localization and exploiting spatial diversity [11–14]. Moreover, GOR maintains high efficiency and scalability since each sensor only needs the local one-hop connectivity. In this paper, our FAGOR uses more candidates as backups and integrates fault activity model into the process of the forwarding candidate selection. For example, as shown in Figure 1, based on distance, energy, trust verification, and delivery quality inside a sensor, each sensor filter is prioritizing to choose a candidate sensor set of the neighbors. These candidates follow the priorities to deliver the packet opportunistically. Malicious sensors (node A and node B) have very low priorities or are even not included in the candidate set according to their direct impacts and indirect impacts.

Network service performance becomes lower when inside intrusions are present since the effective flow gets thinner when misbehaving nodes are on its routines [15, 16]. Therefore, it is necessary to apply rate-control design to complement secure routing and guarantee performance. A popular approach for reliable resource allocation is to design improved optimal flow control (OFC) algorithms, which solve network utility maximization (NUM) problems with constraints on fixed reliability requirements [17–19]. However, these approaches are unable to adopt their resource allocation and fairness dynamically according to the actual-receive rate of each service. We develop a FA-leaky-hop model in which each faulty sensor has potential effects on the resulting data throughput and incorporate the actual-receive rate at wireless hops into OFC approach.

Moreover, when multiple city services, for example, camera monitoring, health surveillance, email, and smart home, are run over a network as shown in Figure 1, the existing OFC approaches usually lead to a serious unfair resource allocation in terms of rates [20]. For example, real-time traffic which has its minimum required rate may get almost zero utility, despite nonzero rates. The utility function conditions of OFC need be relaxed to describe different services regarding heterogeneous traffic types. Based on FA-leaky-hop model, we formulate the problem of allocating rate among multiple services as a lossy flow optimization problem, namely, fault

activity utility OFC, through maximizing the sum of relaxed utilities subject to the network constraints. Considering the existence of faulty sensors, our FA-UOFC algorithm allocates traffic to various services and achieves fairness in terms of actual-receive utility, rather than that in terms of rate or utility. In particular, we define the utility fairness index which could measure the degree of fairness performance based on the achieved throughput in lossy networks and seek to gain its considerable value under our service delivery strategies.

In this article, we investigate multiple city service delivery of joint routing and rate-control that can minimize performance degradation in the event of misbehaving nodes. To the best of our knowledge, we are the first work to address both routing and rate-control for multiple services in WSNs via a fault-dynamic model-based approach. The main contributions of this paper are outlined as follows:

(i) We design a distributed framework of fault activity information at each sensor to locally characterize the impact of the nondeterministic and dynamic faulty behaviors and to incorporate fault activity information into data delivery for multiple city services.

(ii) We propose a fault activity-based geographic opportunistic routing protocol, FAGOR, which combines the direct and indirect impacts of faulty behaviors, to protect against a wide range of attacks.

(iii) We formulate the problem of allocating resources among multiple services in the presence of misbehaving nodes as a lossy flow optimization problem along leaky-hop model. A distributed algorithm, FA-UOFC, is developed to allocate the effective rate properly within the sensor networks and to achieve lossy utility fairness by sources with different traffic types.

(iv) We define a novel index, index of utility fairness, that quantitatively measure the degree of utility fairness among multiple city services in distributed systems.

The rest of the paper is organized as follows. Related work is described in Section 2. We depict our system model in Section 3, and we present methods that allow sensors to establish novel metrics fault activity (FA) according to the impact of misbehaviors in Section 4. In Section 5, we introduce the formulation of a GOR protocol based on FA metrics. In Section 6, we describe the leaky-hop model and formulate the optimal rate-control for multiple services in the presence of misbehaving nodes. The performance of our algorithm is evaluated in Section 7. Finally, we conclude the paper and give directions for future work in Section 8.

2. Related Work

Over the past few years, literatures investigated the multiple city service delivery over wireless networks. A resource management scheme is proposed in [21] to offer the delivery of various city services in the Internet of Things. Tang et al. [22] propose a cross-layer resource allocation model for guaranteeing the QoS requirements of elastic service (audio and video surveillance, habitat monitoring, and real-time traffic monitoring) based on the optimal achievable rate in Cloud Radio Access Network. Spachos et al. [23] design an energy-aware dynamic routing scheme to improve the QoS-aware routing of multimedia traffic by optimizing the selection of the forwarding candidate set. The feasibility of the schemes mentioned above does not consider the existence of malicious nodes, and there is no policy given to defend the misbehaviors of wireless nodes. There exist works that study particular misbehaviors of node-selfishness for multiservice delivery. Luo et al. [24] design an algorithm to select relay nodes in terms of residual energy metrics in WSN-based IoT. The "ground truth" status of each node in [25] is served as virtual credit to encourage data delivery according to its social and QoS behavior. The work in [26] presents a dynamic trust management for secure routing to deal with selfish behaviors and trust-related attacks. Our fault-aware routing and resource allocation scheme extends from these solutions with consideration given to a wider range of misbehaviors on the multiservice delivery in WSNs from the perspectives of both direct-impact factors and indirect impact factors.

Due to the misbehaving nodes' effect on network performance, various defense strategies dealing with the nodes' misbehaviors have been studied for wireless networks. However, most of these works only present countermeasure analysis for different types of faulty nodes and have not considered the uncertainties and dynamics of real environments. Most of the studies assume that the faulty nodes employ a constant strategy that will not change with time. In fact, a faulty node can adopt variable misbehaviors to maximize its intrusion strength [27]. Malicious nodes can be equipped with cognitive technology and can adapt their attacking strategy according to the legitimate users' actions [28]. The attackers decrease their attacks in frequency to disguise themselves and to avoid being detected [29]. Mitchell and Chen [30] characterize a malicious attacker by its capacity to perform random attacks. Similar to [30], our approach works against misbehaving behaviors which may exhibit inconsistent behaviors; a misbehaving node acts as a good node and does not launch attacks at first, in order to gain the trust of other nodes, or, it may perform on-off attacks with a random probability. Our work characterizes the impact of potential dynamic faults and incorporates statistical information into the resource allocation and routing protocols. This assumption not only provides efficient defense against stationary failures but also is suitable for mobile attacks and the uncertain losses from the various environments.

In the reliable routing of WSNs, geographic routing is an attractive approach since no end-to-end route is determined before data delivery [31]. A QoS-aware geographic opportunistic routing, QGOR, is explored in [14] for delivering packets with both time delay and reliability constraints in WSNs. Using location information, Wu et al. [32] design an efficient routing and load balancing algorithm in hybrid VANET. These studies, however, do not consider and respond to location-related attacks. Liu et al. [33] consider the use of the location verification such that neighbors exchange their location information to address a series of location-related attacks. One main limitation of this scheme is that

if the localization mechanism is separated from the routing protocol, the protocol will fail. FAGOR is similar to those schemes in terms of security requirements. FAGOR differs from them in that it uses RSS to detect location information and the verification from the other sensors to identify this type of misbehaviors with possibility.

An optimization problem is first applied to formulate the rate-control stack design of the wireline context by Kelly et al. [34]. This pioneering work was further advanced by studies in cellular wireless networks [35], ad hoc networks [36], and wireless sensor networks [37]. The fundamental assumption of the above research is that each application attains concave utility function and, thus, is only suitable for elastic traffic. It cannot deal with the resource allocation of multiple services in sensor networks where both elastic and inelastic traffic are commonly engaged. Lee et al. [38] show that instability and high network congestion may be caused by the mixing of inelastic and elastic traffic in the absence of appropriate rate controllers. Hande et al. [39] have further derived the sufficient and necessary conditions of system optimality in a mixed-traffic scenario and have proposed a link provisioning method which could potentially be used during the network-planning stage. Alternatively, Wang et al. [20] have developed a new rate-control framework that is able to deal with both elastic and inelastic traffic of multiple services such that the resulting utility is proportional fair. However, these works do not consider the existence of misbehaving nodes and assume that each wireless node is cooperative and well-behaved.

Recently, numerous protocols which maximize the sum of each application's utility by setting fixed reliability constraints have been proposed to allocate the resources of multiple services to provide reliable wireless transmissions [16]. Their works, however, are unable to adapt fairness dynamically in terms of the actual-receive resource of each application. Li et al. [19] incorporate rate, in addition to delay and reliability, into the utility function to support different QoS requirements of various traffic. In our paper, we take a similar approach that the utility is defined to be a function of effective utility received at destination nodes. By means of embodying QoS objectives in the extended utility function, our FA-UOFC is applicable for various services addressing their real utility requirements and improves the utility performance both of inelastic sources and elastic sources.

3. System Model and Assumptions

This section presents the network and the misbehaving-node model handled in this article, as well as the assumptions made in order to design the proposed architecture.

3.1. Network Model. In a smart city, a wireless sensor network involves tiny devices, called sensor nodes $\mathbf{V} = \{1, 2, \ldots, V\}$, which have ability to cater to different applications. These devices are randomly deployed in a city area with a constant size, for example, a smart community containing residential buildings, hospitals, schools, shopping malls, cafes, and banks. Two SNs within the wireless transmission range R can send data and communicate with each other, and any two nodes with a distance greater than R would require a multihop to communicate with each other. A link is denoted as a pair as nodes (i, j), where $i \in \mathbf{V}$ is the transmitter and $j \in \mathbf{V}$ is the receiver. The data collected by sensors is sent to sinks which process data locally or through core networks such as the Internet.

The location of sinks as data, computation, and control center are known in the network. Each sensor knows the geographic coordinate of itself using one of secure localization algorithms [40]. Meanwhile, a sensor can adapt its location information with the help of some trusted mobile anchor nodes in neighbor set, for example, vehicle nodes equipped with GPS.

Due to the broadcast nature of the wireless medium, the transmitters contend in wireless channel capacity for the shared wireless medium if they are within the interference range of each other. Considering the protocol model [41] for successful transmission, the interference among the transmissions is characterized by the interference sets. Since the transmitters included in the interference set share the same common channel capacity, only one of the sensors may transmit over a channel in a time slot. Moreover, since energy is a major concern in WSNs, we assume that sinks are powerful services for collecting data and that other sensors have limited and unreplaceable batteries. We build a power dissipation model to guarantee the operational lifetime of the sensor network in Section 6.

3.2. City Services. WSNs provide a variety of services to city users that will force networks to support heterogeneous traffic. More generally, utilities of multiple city services in a smart city can be categorized as follows in terms of performance goal perspectives [20]:

(i) Elastic utility for traditional data services such as file transfer, mail, and ftp

(ii) Inelastic utility including real-time utility, rate-adaptive utility, and stepwise utility such as video surveillance, real-time monitoring, and teleconferencing

Figure 1 illustrates an example network with five flows s_1 to s_5 of source rates x_1 to x_5, respectively. There are different types of sensors embedded to support city services with different QoS requirements. The utility types of source nodes are given as follows: inelastic utility for the first four source nodes and elastic utility for the fifth source node. Note that, in comparison with other data delivery for elastic traffic, the assumption of mixed traffic in our rate-control model is practical for many smart city applications, such as water consumption, electricity consumption, target tracking, health surveillance, and smart home appliance.

3.3. Fault Activity Information. In this article, we assume that the source nodes have no prior knowledge of the abnormal behaviors of nodes being performed. That is, we make no assumption about the malicious nodes' strategies, misbehaviors' goals, or mobility patterns. We assume that the types of misbehaviors, like failure of internal components or external faults, are unknown to the network.

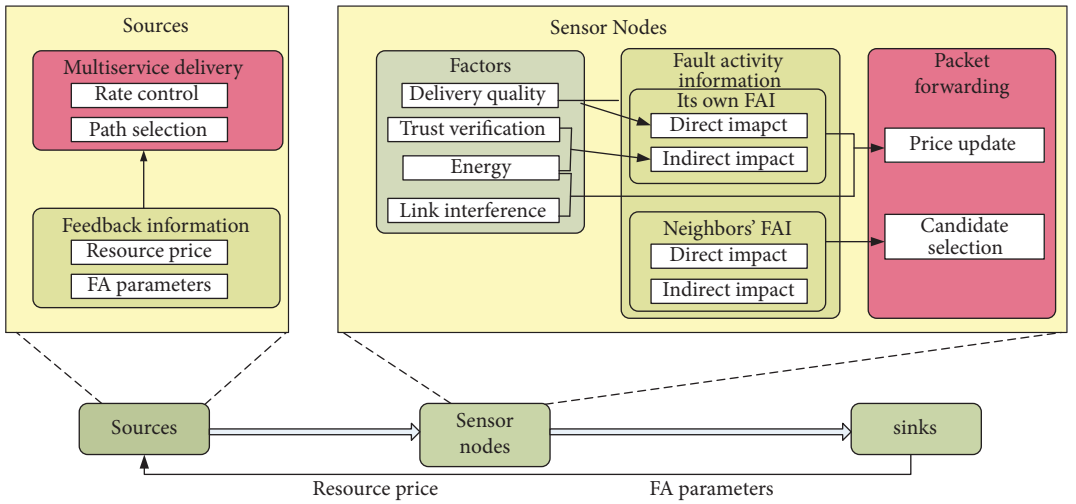


Figure 2: The delivery framework for multiple services based on the fault activity information.

In order to characterize the effect of nodes' misbehaviors on the multiservice delivery, each source must collect information on the impact of the misbehaviors in city parts of networks. However, due to the distributed characteristic of wireless sensor nodes, no central network entity collects the information on the misbehaviors' impact of all sensors and a fully distributed solution is required. Every source/SN should have its own fault activity information (FAI) for both its neighbors' and its own faulty behavior impact. The node FAI at each SN obtains the faulty activity impact of its neighbors and of itself in terms of direct and indirect impacts recommended by the SNs around it. Meanwhile, the direct and indirect impacts are affected by SNs' factors, that is, energy, trust verification, and delivery quality inside a sensor.

When sensor node i delivers multiservices to the sink via multihop communication, there are some candidates based on node i's knowledge of available forwarding neighbors. Nevertheless, since the node misbehaviors may degrade the reliability of the routing path, each hop selects the most reliable one of these candidates in terms of their FAI. Additionally, each sensor node tries to maximize the benefit by sending the feedback signal, the "resource price" determines the cost of consuming limited resources by competing services, to the source. Accordingly, each source is charged the resource price and is then allocated a certain amount of resources for delivering its service. For various types of services or applications, each source is associated with a utility function that reflects how much QoS benefit that source obtains at the allocated transmission rate. Here, the network model of the distributed framework of the candidate selection and rate allocation of the sources is shown in Figure 2.

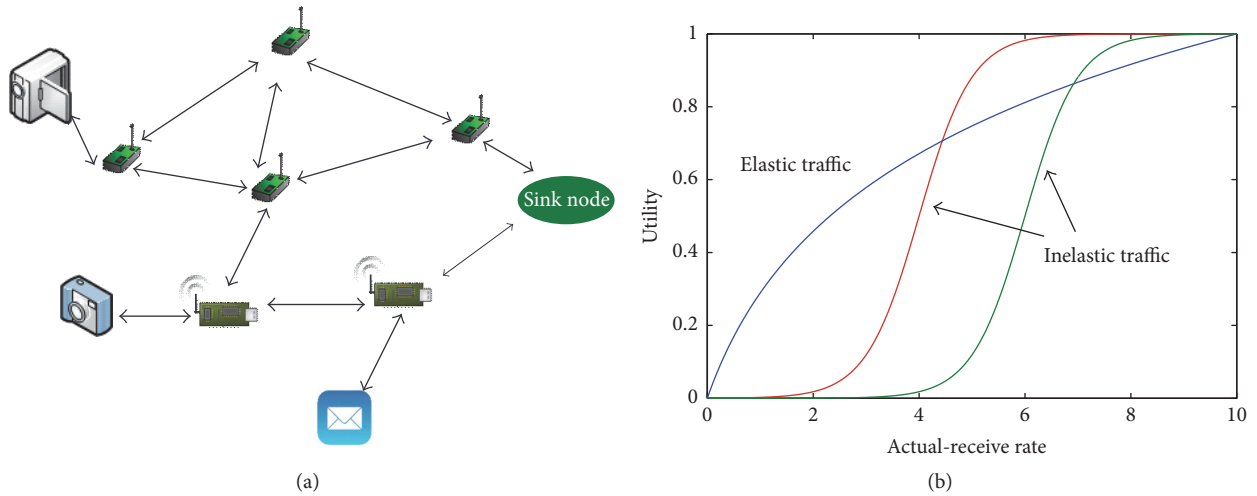
4. Characterizing the Impact of Faulty Activities

In this section, we propose techniques for sensor node estimation and characterization of the impact of faulty activities and for obtaining misbehavior information. Under the distributed framework of the fault activity information (FAI), the FAI of each sensor node consists of two parts: direct impact and indirect impact of misbehaviors on multiservice delivery. Based on FAI, we determine the node-faulty state and get the estimation of FA metric. Each relay sensor should incorporate its neighbors' estimates into its candidate selection for next-hop from its neighbor set. In order for a source node to incorporate the misbehavior impact in the rate-control problem, its own estimation of FA must be recorded in the data packets when the packets arrive at this intermediate sensor and be sent back to the source node when the packets arrive at the sinks.

4.1. Direct-Impact Model

4.1.1. Delivery Quality inside a Sensor. In a smart city, sensors with heterogeneous nature support and forward a mix of elastic and inelastic traffic. With the existence of misbehaving sensors along routing paths, the data rate of a flow gets thinner and thinner and the actual-receive rate at the sink is considerably lower than that at the source. Figure 3 shows the utility obtained by elastic and inelastic applications at different actual-receive rates. If an elastic service gets a rate slightly greater or lower than their minimum required rate, inelastic applications get zero utility. Therefore, the quality of delivery inside a sensor is a significant factor for utility of multiple services.

Although a faulty node may perform various behaviors, any good node exhibits the same behavior: delivering packets correctly. Similar to the approach in [42], we use the ratio of packets successfully delivered compared to those sent (packets may be corrupt even if received) in order to characterize the delivery quality inside a sensor. During a certain period $[t - T, t]$, each node (sender) enters the promiscuous mode and checks whether the packet is actually forwarded by its selected nodes. Additionally, it can record in the neighbor list the running average number $NR_i[t - T, t]$ of packets sent to node i and the running average number $NV_i[t - T, t]$ of valid



(a) (b)

Figure 3: Utility of elastic and inelastic services.

packets. Each sensor is aware of the delivery quality values of any node i and of its one-hop neighbors for the period $[t-T,t]$, denoted as $PR_i([t-T,t])$:

$$PR_i([t-T,t]) = \frac{NV_i[t-T,t]}{NR_i[t-T,t]}. \quad (1)$$

4.1.2. Energy. If some sensors malfunction due to the lack of energy, this degrades the overall network efficiency and performance. E_i is denoted as the remaining energy of node i. Let e_s, e_t, and e_r be the energy consumed in the sensing, transmitting, and receiving for one data packet per unit time.

$$E_i = \begin{cases} e_s + e_t & \text{if flow } s \text{ starts from node } i \\ e_t + e_r & s \in S(i) \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

In order to update the direct-impact metric, the location beacon of one-hop neighbors is extended to apply an additional field of remaining energy $E_i(t)$. We can use $PR_i([t-T,t])$ and E_i to update the estimate $DI(t)$ at the end of the time interval. In order to balance the stability and the accuracy of the estimation results, we update the estimation $DI(t)$ through iterations:

$$DI(t) = \kappa(\alpha DI(t-T) + (1-\alpha) PR_i([t-T,t])) + (1-\kappa) E_i(t), \quad (3)$$

where $0 < \alpha \leq 1$ is the parameter that controls the preference between current and historic samples and $0 < \kappa \leq 1$.

4.2. Indirect Impact Model

4.2.1. Trust Verification. In smart environments, the network also has one or more malicious users that control a number of malicious colluders. All colluders may cooperate with each other and turn their partner into an inside faulty node. During the initial stage or under a random attack strategy, these malicious nodes do not immediately launch packet dropping behaviors, and they modify their transmission power to disguise themselves. Hence, the impact of the disguised nodes' misbehavior is indirect on packet delivery from the perspective of the network, and a validation metric can be applied to distinguish malicious nodes with the voting-based scheme.

To keep consistency, we follow the assumption and variable definitions about GOR in [43]. Each node periodically broadcasts the location beacon with the location information to its one-hop neighbors. After receiving the beacon from node A, a neighbor B verifies the location information in terms of the received signal strength. RSS is given by the following [44]:

$$\text{RSS}_{AB}\ (\text{dBm}) = P_t - p_0 - 10\beta \lg\left[\frac{d_{AB}}{d_0}\right] + x, \quad (4)$$

where P_t is the node's transmission power in dBm and β is the path loss factor. Here, p_0 is the path loss at the reference distance d_0 and x is a random variable. However, if the RSS is susceptible, the above approach will lead to high false negatives against location-related attacks. Based on (4), the distance is estimated as $D_{AB} = d_{AB}(1 \pm \rho)$, where ρ is the measurement error. To reduce the effect of the disguised nodes, node A requires collecting more RSS value from the information of its common neighbors. We denote $\mathbf{H} = N^{(A)} \wedge N^{(B)} = \{H_1, H_2, \ldots, H_k\}$ as the intersection of A's neighbor set and B's neighbor set. A neighbor node R_j is selected by B to find the difference of the RSS value of the sender in H (e.g., node H_j). Even though the transmission power may be modified, the difference $R_{BR_j}^{H_j}$ is found to be constant [45]:

$$R_{BR_j}^{H_j} = \frac{\text{RSS}_{H_jB} - \text{RSS}_{H_jR_j}}{10\beta} = \lg\frac{d_{H_jR_j}}{d_{H_jB}}. \quad (5)$$

As either the node H_j or the chosen neighbor node R_j may use forged information of this distance value, $D_{H_jR_j}$ or

D_{H_jB} are used to replace the value of $d_{H_jR_j}$ and d_{H_jB}. We can get the inequality from (5):

$$\lg\frac{D_{H_jR_j}}{D_{H_jB}} + \lg\frac{1-\rho}{1+\rho} \le R_{BR_j}^{H_j} \le \lg\frac{D_{H_jR_j}}{D_{H_jB}} + \lg\frac{1+\rho}{1-\rho}. \quad (6)$$

Following this method, we can obtain $(R_{BR_j}^{H_1}, R_{BR_j}^{H_2}, \ldots, R_{BR_j}^{H_c})$ for other nodes in set **H**. In this round, two disguised nodes H_m and H_i are identified with R_j, provided that

$$R_{BR_j}^{H_i} + \lg\frac{1-\rho}{1+\rho} \le R_{BR_j}^{H_m} \le R_{BR_j}^{H_i} + \lg\frac{1+\rho}{1-\rho}. \quad (7)$$

With node B's neighbor nodes as reference nodes, each H_i belonging to H can be identified using this method. During the time period $[t-T, t]$, there are $q_i([t-T, t])$ disguised nodes that are faked by actually one node in a round and $f_{H_i}([t-T, t])$ rounds of the entire $m_{H_i}([t-T, t])$ rounds in the calculation. The estimate value $DS_{H_i}(t)$ of the possible disguiser H_i can be obtained by

$$\begin{aligned} DS_{H_i}(t) &= \gamma DS_{H_i}(t-T) \\ &\quad + (1-\gamma)\frac{1}{q_{H_i}([t-T,t])}\left(1 - \frac{f_{H_i}([t-T,t])}{m_{H_i}([t-T,t])}\right). \end{aligned} \quad (8)$$

An attacker can launch a spoofing attack by sending forged location beacons to attract SNs to choose one of them as the next-hop. In this paper, the FAI management makes use of the RSS to verify SNs' location and to address the location-related attacks by offering nodes the location with possibility. Based on the collected RSS values, we can compute the values $(R_{BH_1}^A, R_{BH_2}^A, \ldots, R_{BH_k}^A)$ for the set **H** whose size is k, where $R_{BH_i}^A = (\mathrm{RSS}_{AB} - \mathrm{RSS}_{AH_i})/10\beta = \lg(D_{AH_i}/D_{AB})$. Then the following inequality can be provided to decide whether node A is marked as a successful validation:

$$\lg\frac{d_{AH_i}}{d_{AB}} + \lg\frac{1-\rho}{1+\rho} \le R_{BH_i}^A \le \lg\frac{d_{AH_i}}{d_{AB}} + \lg\frac{1+\rho}{1-\rho}, \quad (9)$$

where d_{AH_i} and d_{AB} are the position announced in the received location beacon. If the inequality is satisfied, it means that node A with one neighbor $H_i \in \mathbf{H}$ can be marked as a successful validation, and $M_{H_i} = 1$. Otherwise, $M_{H_i} = 0$. We can obtain the ratio of successful validation of node A:

$$LC_A(t) = \frac{1}{k}\sum_{i=1}^{k} DS_{H_i}(t)\, M_{H_i}. \quad (10)$$

Furthermore, we introduce the indirect impact metric to address issues of location-related attacks. In order to gain the trust of other nodes, some malicious sensors claim themselves as legitimate nodes but transmit beacon messages containing false location information to confuse other sensors. Each network node may obtain the verification information of its candidates indirectly received from its neighboring nodes. Additionally, the impact of these disguised nodes' misbehavior which pollutes the network system with bogus information is indirect on packet delivery from the perspective of the network. We get the expression of indirect impact metric of node A:

$$IDI_A(t) = \delta_1 DS_A(t) + \delta_2 LC_A(t), \quad (11)$$

where $\delta_1 + \delta_2 = 1$ and $0 < \delta_i < 1$ which is the coefficient factor. The indirect impact metric of each node's one-hop neighbors can be calculated in terms of information in the beacon. To reduce the bandwidth consumption caused by beacon exchange, it is not necessary to contain the neighbor information in the beacon unless the information is changed.

4.3. Fault Activity Metric Based on Determining Node State. Due to the uncertainty in the faulty impact, we model the direct impact and the indirect impact as random processes and allow the sensor nodes to collect empirical data for characterizing the process. In order to identify the faulty state of each node, we design an impact metric which enables each node to measure faulty impact for both its own faulty impact and its neighbors' faulty impact based on its knowledge of available one-hop neighbors. The total impact value for node A can be given by

$$I_A(t) = \epsilon DI_A(t) + (1-\epsilon)\, IDI_A(t), \quad (12)$$

where ϵ is the factor with $0 < \epsilon \le 1$. Then we define the novel faulty state and FA metric as follows.

Definition 1 (the node-faulty state). $\Lambda_i(t_0)$ denotes the faulty status in node i at time t_0, where $\Lambda_i(t_0) = 1$ indicates that the node i is faulty where $I_i(t_0) \le I_0$; otherwise, $\Lambda_i(t_0) = 0$ indicates that node i is not faulty.

To determine the node-faulty state, we can use a heuristic approach to test whether the current node is experiencing "being faulty condition" in which the impact metric drops below a certain threshold. Any node whose impact metric is below the threshold can be regarded as a faulty node since we are unable to accomplish our objectives efficiently. We suppose that each node i updates DI_i and IDI_i after each update period of T seconds and estimates the FA metric after each update calculation period of $T_s \gg T$ seconds. Next, we define the FA which is the time that faulty nodes spend in each state per unit time.

Definition 2. The FA for node-faulty state denoted by A_i is the fraction of time during period $[t-T_s, t]$ for which the node i is in the state Λ_i, that is, $A_i = (T/T_s)\int_{t-T_s}^{t} \Lambda_i(x)dx$.

To facilitate observation, we illustrate an example of converting the impact value of a sensor node A (as shown in Figure 4) into the faulty state with I_0 being 0.6 in Figure 5 and the value of fault activity in Figure 6. Once we obtain the estimation of FA, we can get the fault-statistical information for routing path selection and resource allocation.

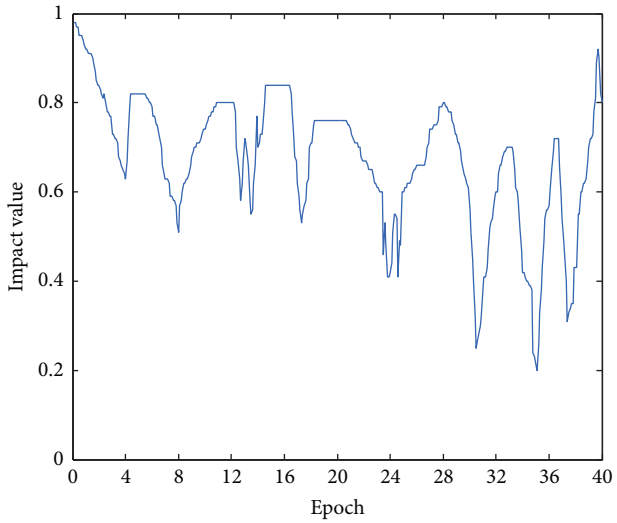


Figure 4: Impact value of a sensor node.

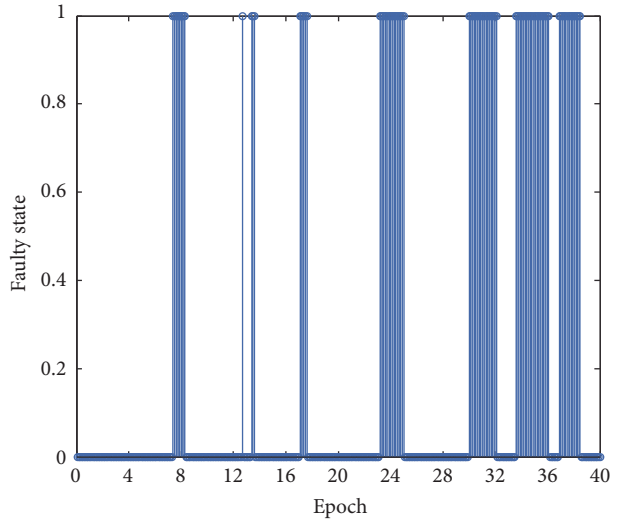


Figure 5: Distribution of faulty state.

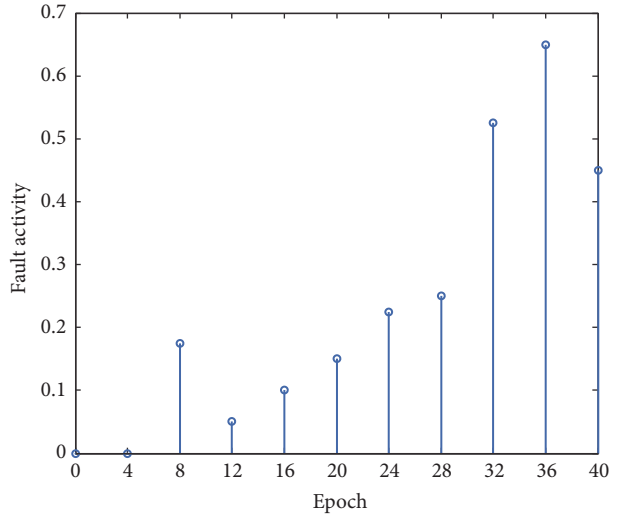


Figure 6: Estimation of FA.

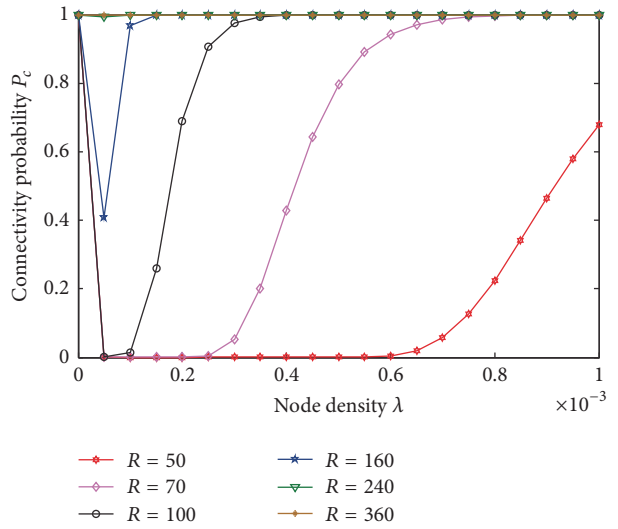


Figure 7: Connectivity probability with $S = 1000 * 1000\,\mathrm{m}^2$, $2 * 10^{-5} \leq \lambda \leq 1 * 10^{-3}$, and $50\,\mathrm{m} \leq R \leq 360\,\mathrm{m}$.

5. Fault Activity Geographic Opportunistic Routing Algorithm

In this section, a geographic routing protocol on fault activity metric is presented, providing methods for sensors to choose the candidates based on impact caused by faulty behaviors. FA-GOR selects more forwarding candidates based on the routing metric of available next-hop forwarders.

Before presenting our routing algorithm, we first discuss an intrinsic nature of WSNs that can support our idea: network connectivity. When sensors are distributed in area S randomly, the process that there are n sensors in an arbitrary area U is modeled according to Poisson distribution [40]:

$$P\{|N_n| U = n\} = \frac{(\lambda U)^n}{n!} e^{-\lambda U}, \tag{13}$$

where λ denotes node density, $|N_n|$ is the cardinality of N_n, and $\lambda = |N_n|/U$. In order to describe the full connection probability P_c, we first calculate the probability P_{iso} that no link exists between sensor N and other nodes:

$$P_{\mathrm{iso}} = P\{|N_n| \pi = R^2 0\} = e^{-\lambda \pi R^2}. \tag{14}$$

In terms of the isolation probability P_{iso}, the full connection probability is given by the following [46]:

$$P_c \geq e^{-\lambda S P_{\mathrm{iso}}}. \tag{15}$$

Figure 7 shows that when λ and R are set as proper values, the expected fully connected can be achieved in a WSN.

Assuming that $\mathrm{Dist}(y, \mathrm{Dest})$ is denoted as the distance from sending node y to the sink (denoted as Dest) and $\mathrm{Dist}(v, \mathrm{Dest})$ is denoted as the distance from its neighbor $v \in N^{(y)}$ to the sink, we have the routing metric for the forwarding candidates as follows:

$$\mathrm{metric}_{yv} = \vartheta \left(1 - \frac{\mathrm{Dist}(v, \mathrm{Dest})}{\mathrm{Dist}(y, \mathrm{Dest})}\right) + (1 - \vartheta)(1 - I_v), \tag{16}$$

Require: $v \in N^{(y)}$, the neighbor set of node y
Ensure: the next forwarder n
(1) start a retransmission timer;
(2) select the forwarding set $F^{(y)}$ including g candidates from neighbor nodes $N^{(y)}$, $F^{(y)} = \emptyset$, $g = 0$;
(3) **for** each node $i \in (N^{(y)} - F^{(y)})$ **do**
(4) **if** $\text{metric}_{yi} = \max\{\text{metric}_{yj}\}$, $\forall j \in (N^{(y)} - F^{(y)})$ and $n \leq g$ **then**
(5) add i to $F^{(y)}$; $g++$;
(6) **end if**
(7) **end for**
(8) prioritize the forwarder set using metric;
(9) broadcast the data packets;
(10) **for** each node $i \in F^{(y)}$ **do**
(11) receive the data packet;
(12) check the sender ID and start a timer and $\text{time}(i) = \kappa/\text{metric}_{yi}$, where κ is a constant;
(13) **end for**
(14) **if** node n which obtains the highest priority receives the data packet correctly **then**
(15) reply an ACK to notify the sender as well as other candidates to cancel their timers;
(16) **else**
(17) **if** the priority timer expire **then**
(18) set $n = n'$, node n' has the lower-priority;
(19) goto 14;
(20) **end if**
(21) **end if**
(22) **if** no forwarding candidate has successfully received the packet **then**
(23) **if** the retransmission timer does not expire **then**
(24) goto 2;
(25) **end if**
(26) **end if**
(27) return

Algorithm 1: FAGOR algorithm.

where $\vartheta \in (0, 1]$ is the constant weight indicating the relative preference between distance and fault impact value I_v. Each next-hop forwarder is assigned with its priority based on the metric value of (16).

We introduce the FAGOR algorithm to select the next relay node following the assigned priority in forwarder set F to relay the packets. Algorithm 1 depicts the pseudocode of FAGOR algorithm.

Our FAGOR could defend against a wide range of misbehaviors. For example, in Figure 8, as one candidate of node B's next-hops, node A lies about its location and associates with disguisers (H_4–H_7) as its colluders. The mutual neighbors of A and B, H_1–H_7, need to report their RSS values related to A to B and work based on majority voting. B could choose reference nodes from $N^{(A)} \wedge N^{(H)}$ to verify the validity of the voters. Node R sends the estimate value DS_{H_i} about H_4–H_7 to node B by (8). Node B calculates LC_A to incorporate it into indirect value of node A. Finally, node A is found as being faulty state during a period and could not be selected into the routing path.

6. Fault Activity Utility-Based Optimal Flow Control Approach

In this section, we present a leaky-hop model which explicitly takes account of faulty activities and then present fault activity-based utility optimal flow control (FA-UOFC) based on the leaky-hop model. One underlying assumption in the utility framework of rate control is that the same flow is present at all the hops along the route. In hostile environments, however, the data rate x_s of a given flow s becomes thinner along its path. Due to potential faulty behaviors on each node, all data deliveries are not successful.

6.1. Leaky-Hop Model. In Section 4, A_i is denoted as the fraction of time during the unit period for which node i exhibits misbehavior, while $1 - A_i$ is the time fraction during which node i accomplishes its communication effectively as a good node. A_i characterizes the probability of faulty behaviors over single hop. At a link (i, j) with transmission rate $\sum_{s \in S(i,j)} x_s$, since data is only received correctly on $1 - A_i$

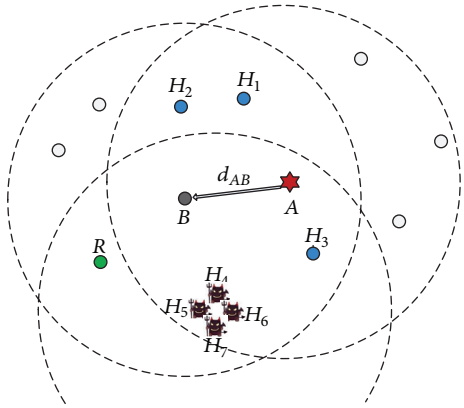


Figure 8: An illustration for misbehaving nodes.

from hop i, the correctly received data rate x_j' at hop j is presented by

$$x_j' = (1 - A_i) \cdot \sum_{s \in S(i,j)} x_s. \tag{17}$$

For path R_s traversing multiple hops, the end-to-end packet success ratio for path R_s is given by

$$\gamma_s = \prod_{(i,j) \in R_s} (1 - A_i). \tag{18}$$

R_s^i is denoted as the subpath of R_s between source S and the intermediate node i, and $\overline{R}_s^i$ is denoted as the subpath of R_s between the intermediate node i and the sink node of R_s. For subpath R_s^i of a data flow, the data delivery probability at leaky-hop i is given by $\gamma_s^i = \prod_{(i,j) \in R_s^i} (1 - A_i)$. It can be seen that the data rate of a given flow becomes "thinner and thinner" at each hop along its routing path, and we call the flow traversing every potential misbehaving hop to be a leaky-hop flow. We define goodput x_s' of flow s as the data rate received correctly at the sink [47]. Therefore, in the presence of misbehaving nodes, $x_s' = \gamma_s x_s$.

An example leaky-hop model is described in Figure 9. Flow 1 traverses along four leaky-hops: n_1, n_3, n_4, and n_6. Flow 2 traverses along three leaky-hops: n_2, n_3, and n_5. The goodput of flow 1 at the destination is $(1 - A_{n_1})(1 - A_{n_3})(1 - A_{n_4})(1 - A_{n_6})x_1$. It can be seen that the data rate of a flow becomes lower and lower along multiple hops. For example, $\gamma_1^{n_1} x_1 \to \gamma_1^{n_3} x_1 \to \gamma_1^{n_4} x_1 \to \gamma_1^{n_6} x_1$. There may exist different data delivery probabilities at a leaky-hop for different data flows. The leaky-hop n_3 for flow 1 and flow 2 has different data delivery probabilities: $\gamma_1^{n_3} = (1 - A_{n_3})\gamma_1^{n_1}$, $\gamma_2^{n_3} = (1 - A_{n_3})\gamma_2^{n_2}$. We call a potential faulty node on the routing path of flow s to be a leaky-hop for flow s.

The resource allocation problem in WSNs gives rise to many new challenges. Among the many unique characteristics of WSNs, we focus on two constraints in our formulation. Due to the broadcast nature of the wireless medium, all transmissions are not successful and the transmitters contend with each other in the broadcast domain. To apply the constraint of contention regions, we use the contention set concept from [48]. The contention set Ω is denoted as the subset of links belonging to a contention region that, at most, one link in Ω can transmit in each time slot successfully. Let $\Omega_{(i,j)}$ be the contention link set of link (i, j). If user s transmits over link (i, j), other flows in the contention set $\Omega_{(i,j)}$ cannot transmit packets simultaneously. Let $c_{(i,j)}$ be the capacity of link (i, j). We incorporate the node-faulty activity statistics into the link capacity constraint generation. Due to leaky-hops along the routing path, the flow rate is potentially reduced at each of the receiving hops as packets are lost. The availability metric in Definition 2 means the fraction of time for which the immediate sensor delivers packets correctly. The stochastic capacity constraint on the total flow rate traversing a link (i, j) is given by

$$\sum_{(i',j') \in \Omega_{(i,j)}} \sum_{s \in S_{(i',j')}} \frac{\gamma_s^i x_s}{c_{(i,j)}} \le 1. \tag{19}$$

Another major point in WSNs is the energy constraint caused by the energy consumption of sensing, transmitting, receiving, and relaying data. Let B_i denote the initial amount of initial battery (energy) at node i, $i \in N$.

We also incorporate the FA statistics into the energy constraint, in which the power consumption of each node i should not exceed the maximum allowed power generation $p_i^{\max}$:

$$(e_t + e_r) \sum_{s \in S(i)} \gamma_s^i x_s + (e_s + e_t) \lambda_i \le p_i^{\max}, \tag{20}$$

where $\lambda_i = \gamma_s^i x_s$, if flow s starts from sensor node i; otherwise, $\lambda_i = 0$. For a prespecified lifetime, T_d, the maximum node power consumption $p_i^{\max} = B_i/(T_d - \tau p_{\text{idle}})$, where τ and p_{idle} are the duty cycle and energy consumed in the idle state per unit time.

6.2. FA-UOFC for Multiple Services. For wireless sensor networks in a smart city, many different types of sensor are emerging to present numerous applications that exhibit different utility behaviors. Similar to [20], we observe that the operations of the data gathering involve both inelastic and elastic traffic. In order to support the multiple types of traffic, the flow control strategy should have the ability to allocate traffic rates properly in order to balance the performance for different applications. We will adopt the rate-control protocol, newly developed by Wang et al. [20], for handling elastic and inelastic traffic. When each source s transmits at rate x_s, it attains a utility $U_s(x_s)$. The utility function $U_s(\cdot)$ is assumed to be continuous, strictly increasing, and bounded in the interval $[m_s, M_s]$. We define a "pseudoutility" $u_s(x_s)$ as

$$u_s(x_s) = \int_{m_s}^{x_s} \frac{1}{U_s(y)} dy, \quad m_s \le x_s \le M_s. \tag{21}$$

In order to provide a good performance balance for different applications in sensor networks, the flow control can be generalized to obtain new problem formulations, namely, utility optimal flow control (UOFC), which maximizes the

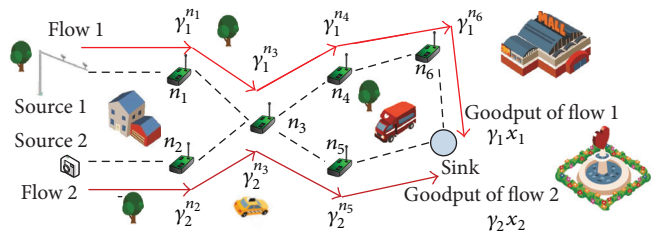


Figure 9: An example network with leaky-hop flows.

sum pseudoutility under the contention constraint [41] and the energy constraint.

At the sink of flow s, the correctly received data rate can be represented as $\gamma_s x_s$. The optimization problem introduced previously can be presented as a new formulation:

$$\begin{aligned} \text{Problem: max} \quad & \sum_{s\in S}\left(\int_{m_s}^{\gamma_s x_s}\frac{1}{U_s(y)}dy\right) \\ \text{s.t.:} \quad & \sum_{(i',j')\in\Omega_{(i,j)}}\sum_{s\in S_{(i',j')}}\frac{\gamma_s^i x_s}{c_{(i,j)}} \le 1 \\ & (e_t+e_r)\sum_{s\in S(i)}\gamma_s^i x_s + (e_s+e_t)\lambda_i \\ & \le p_i^{\max}. \end{aligned} \tag{22}$$

Since the objective function $U_s(\cdot)$ is nonnegative, continuous, and strictly increasing (not concave), the "pseudoutility" $\int_{m_s}^{\gamma_s x_s} 1/U_s(y)dy$ must be a strictly increasing concave function. Therefore, with linear, separable, convex, and compact constraints, the optimization problem in (22) has a unique optimal solution.

In the following, we use Lagrangian dual method and develop a rate-control algorithm. First, we form the Lagrangian as follows:

$$\begin{aligned} L(x',\underline{\lambda},\overline{\lambda}) = & \sum_{s\in S}\left(\int_{m_s}^{\gamma_s x_s}\frac{1}{U_s(y)}dy\right. \\ & -\gamma_s^i x_s\left(\left(\sum_{(i,j)\in L(s)}\sum_{(i',j')\in\Omega_{(i,j)}}\underline{\lambda}\right)+(e_r+e_t)\sum_{i\in N(s)}\overline{\lambda}\right. \\ & \left.\left.+(e_s+e_t)\iota_s\right)\right)+\sum_l\underline{\lambda}c_l+\sum_{i\in N}\overline{\lambda}p_i^{\max}, \end{aligned} \tag{23}$$

where $\underline{\lambda}=[\underline{\lambda}_1,\underline{\lambda}_2,\ldots,\underline{\lambda}_L]^T$, $\overline{\lambda}=[\overline{\lambda}_1,\overline{\lambda}_2,\ldots,\overline{\lambda}_S]^T$, and $u=(\underline{\lambda},\overline{\lambda})$ are all nonnegative. $\iota_s=\overline{\lambda}$, assuming flow s starts from node n. The objective function of dual problem is given by

$$\min_{\underline{\lambda},\overline{\lambda}} \quad D(\underline{\lambda},\overline{\lambda}) = \min_{\underline{\lambda},\overline{\lambda}\ge 0} \quad \max_{x'} L(x',\underline{\lambda},\overline{\lambda}). \tag{24}$$

We use the gradient method to solve the above dual problem. The Lagrangian multipliers for the dual can be updated as follows at each iteration t:

$$\begin{aligned} \underline{\lambda}_{(i,j)}(t+1) = & \left[\underline{\lambda}_{(i,j)}(t)+\varphi\left(\sum_{(i',j')\in\Omega_{(i,j)}}\sum_{s\in S_{(i,j)}}\left(x_s\gamma_s^i\right)\right.\right. \\ & \left.\left.-c_{(i,j)}\right)\right]^+, \end{aligned} \tag{25}$$

$$\begin{aligned} \overline{\lambda}_{(i)}(t+1) = & \left[\overline{\lambda}_{(i)}(t)\right. \\ & +\varphi\left(\left((e_t+e_r)\sum_{s\in S(i)}x_s+(e_s+e_t)\lambda_i\right)\gamma_s^i\right. \\ & \left.\left.-p_i^{\max}\right)\right]^+, \end{aligned} \tag{26}$$

where $\varphi>0$ is a small step size, and $z^+=\max\{0,z\}$. Here, $\underline{\lambda}_{(i,j)}$, $(i,j)\in L$, can be considered the price for using the resource of contention set $\Omega_{(i,j)}$. Similarly, $\overline{\lambda}_{(i)}$, $i\in N$, can be interpreted as the price for using energy at sensor node i. Given these two prices, each flow s, $s\in S$, adopts its rate according to

$$x_s(t+1) = \left[u_s'^{-1}(\lambda^s(t))\right]_{m_s}^{M_s}, \tag{27}$$

where $[z]_b^a=\min(\max(z,a),b)$, $u_s'^{-1}$ is the inverse of u_s', and (27) can be replaced as follows:

$$x_s(t+1) = U_s^{-1}\left(\left[\frac{1}{\lambda^s(t)}\right]_{U_s(m_s)}^{U_s(M_s)}\right), \tag{28}$$

where $\lambda^s(t)=\sum_{(i,j)\in L(s)}\underline{\lambda}_{(i,j)}(t)+(e_r+e_t)\sum_{i\in N(s)}\overline{\lambda}_i(t)+(e_s+e_t)\iota_s(t)$. Hence, we propose Algorithm 2 based on the problem formulation of fault activity-based utility optimal control.

Our algorithm can be carried out in a distributed manner by message exchange in the network, as shown in Figure 10. To implement our scheme, no node in the network needs to know global information nor the individual variables of algorithm. The information needs to be updated by the receiving node and to be sent via piggybacking.

At each time $t = 1, 2, \ldots,$
(1) Update source rate: Each source node s calculates the source rate $x_s(t + 1)$ for the next period according to Eq. (28);
(2) Update resource prices: Using the information of aggregated transmission rate, link (i, j) computes a new sole contention price $\underline{\lambda}_{(i,j)}(t + 1)$ according to Eq. (25) and node i computes a new energy price $\overline{\lambda}_i(t + 1)$ according to Eq. (26);
(3) Deliver information towards the sink: Sensor node i adapts the contention price $\underline{\lambda}_{(i,j)}(t)$ and the energy price $\overline{\lambda}_i(t)$ along the path, and propagates towards the sink;
(4) Feedback message from the sink: The sink feedbacks the FA parameter and the aggregated resource price to the source via the reverse path.

Algorithm 2: FA-UOFC algorithm.

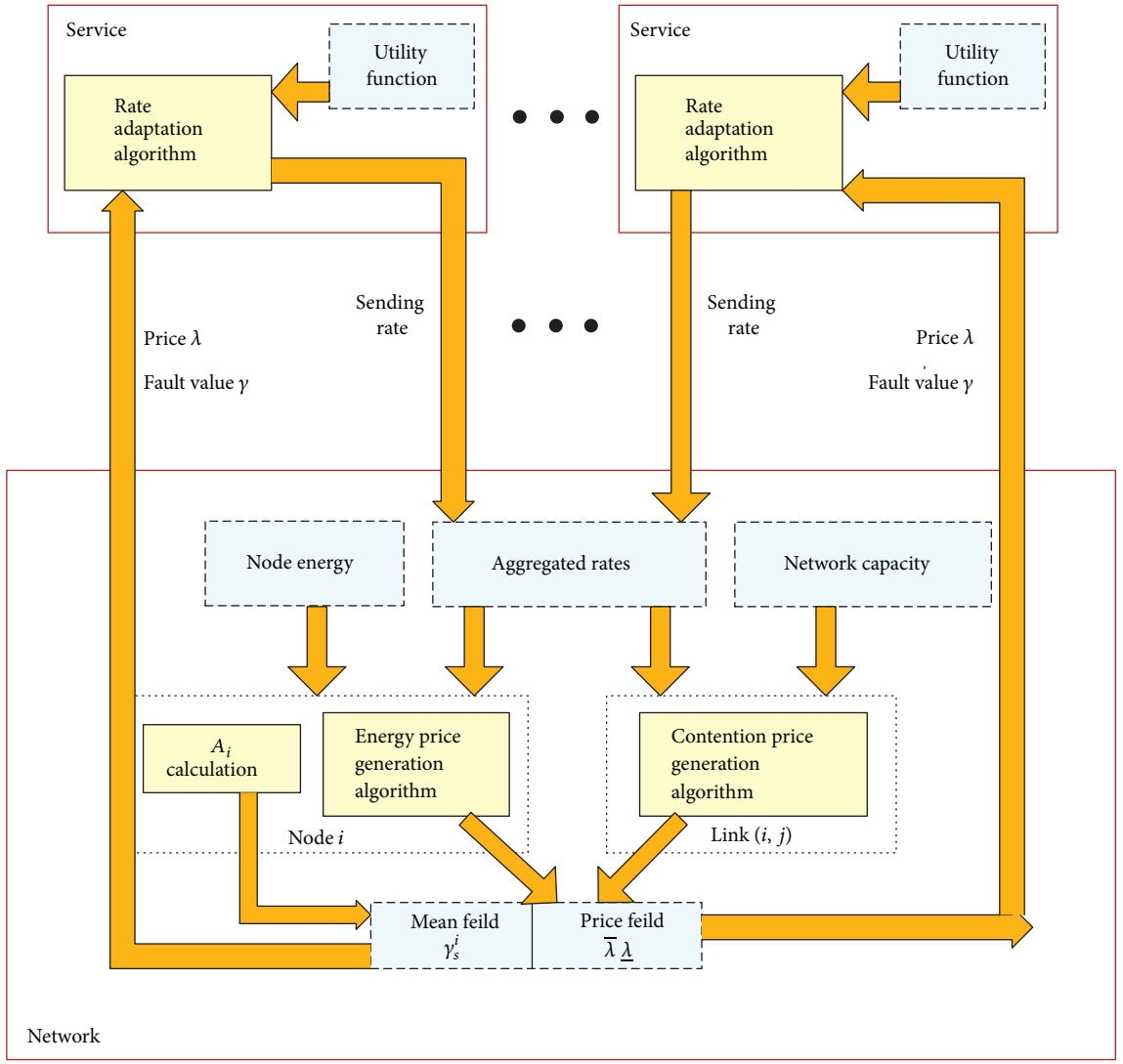


Figure 10: System model for Algorithm 2.

First, each sensor node estimates and updates the resource price locally, the fault activity information of its neighbors, and its own fault activity information; then we apply two additional header fields, mean field and price field, to both data packets and control packets. When a new packet arrives, the updated FAI is multiplied together and the local prices are added to the price of the packets that arrive from the upstream node. When the packet arrives at the sink, values of the two fields will be feedback to the source node by the acknowledgement packet.

Second, when the packet arrives at the sink, the aggregated FAI and resource prices will be piggybacked to the source node in the acknowledgement packet.

Third, each node can construct its local contention set by exchanging information from neighbors instead of knowing the entire network topology.

Hence, the total number of additional exchange operations is within $O(LN)$, where N is the number of source S' routing paths and L is the number of network's links. The proposed fault activity utility optimal flow control algorithm is practical and realizable in WSNs.

6.3. Utility Fairness. The goal of our rate-control approach is to able to maintain an acceptable level of service degradation, including effective network throughput and fairness, in the presence of misbehaving nodes. In this section, we establish the existence and uniqueness of a utility fair solution with the presence of misbehaving nodes and define a novel index, utility fairness index, which quantitatively measures the degree of utility fairness in distributed systems.

Considering the performance of different services, the utility OFC (UOFC) with the resource constraints in WSNs allocates flow rates of different applications according to their utility requirements, and, what is more, the optimization approach yields utility fairness [20]. In WSNs without faulty nodes, the set of goodput rate vector X for each flow s that satisfies the resource constraints in problem (22) with $\gamma_s^i = 1$ for $i \in N$ is called the rate region $X(c, s, 1)$. In hostile environments, the set of goodput vector X' that follows from problem (22) with $\gamma^i \neq 1$ is denoted as $X(c, s, \gamma)$. It is clear that $\gamma_s^i \leq 1$ and that $X(c, s, \gamma) \subseteq X(c, s, 1)$.

When the rate-control Algorithm 2 with $A_i = 0$ leads to equilibrium $(x^*, \underline{\lambda}^*, \overline{\lambda}^*)$ at convergence, the pseudoutility function $u(x)$ is maximized within the feasible solution. Here we can employ both a utility proportional fairness as described in [20] and utility max-min fairness proposed in [48]. For any other feasible allocation $x \neq x^*$, if $\sum_{s\in S}(\partial u(x_s^*)/\partial x_s)(x_s - x_s^*) = \sum_{s\in S}((x_s - x_s^*)/U_s(x_s^*)) \leq 0$, the source rate allocation $X^* = [x_1^*, x_2^*, \ldots, x_s^*]^T$ is utility proportionally fair. $U(x)$ is the strictly concave function; the strict inequality holds and meets the utility proportional fairness definition. Therefore, the source rate allocation in Algorithm 2 with $\gamma_i = 1$ is utility proportionally fair. To achieve utility max-min fairness, we give a new distributive flow control algorithm. If the aggregate price of Algorithm 2 is replaced with $\lambda^s(t) = \max\{\max_{(i,j)\in L(s)}\underline{\lambda}_{(i,j)}(t), \max_{i\in N(s)}\overline{\lambda}_i(t)\}$, which is the maximum of the contention prices and the energy prices along the path, the updated algorithm could provide a utility max-min fair allocation among all data flows.

6.3.1. Utility Fairness of $X(c, s, \gamma)$. We relate the arguments on utility OFC based on the leaky-hop model to a case without leaky-hop by proving a continuity property of fair allocation as γ_i approaches 1. Let the ratio of node-faulty activities drop to zero: $\lim_{k\to\infty}\min_{(i,j)\in R_s}\gamma_i^k = 1$. Then the rate regions in WSNs containing faulty nodes converge the rate regions in the corresponding WSNs without faulty nodes, and utility fair solution converges to the corresponding utility fair solution without faulty nodes [47].

The goal of our rate-control approach is to be able to maintain an acceptable level of service degradation, including effective network throughput and fairness, in the presence of misbehaving nodes. In this section, we establish the existence and uniqueness of a utility fair solution with the presence of misbehaving nodes and define a novel index, utility fairness index, which quantitatively measures the degree of utility fairness in distributed systems.

In the homogeneous traffic context, Jain et al. [49] propose a quantitative measure called Index of Fairness to tell how far the resource allocation is from equality. With considering QoS requirements of different applications, it may be undesirable to allocate resources simply according to conventional measurements such as Index of Fairness [49]. Hence, we define a novel index, *index of utility fairness* $f(x)$, which measures the utility fairness of various applications and addresses their utility requirements:

$$f\left(x'\right) = \frac{\left(\sum_{s=1}^{n} u\left(x_s'\right)\right)^2}{|N|\sum_{s=1}^{n} u\left(x_s'\right)^2}, \tag{29}$$

where x_s' is the goodput of flows and $|N|$ is the number of flows in WSNs. This index measures the "equality" of user utility allocation. If all sources get the same amount of utility, that is, if $u(x_i')$ are all equal, then the utility fairness index is 1. As the disparity increases, the utility fairness decreases and is near 0 as only a selected few users will be favored. A higher value of $f(\cdot)$ means a higher degree of utility fairness.

7. Performance Evaluations

In this section, we conduct simulation experiments to evaluate the performance of the proposed FAGOR protocol and FA-UFOC scheme when misbehaving nodes exist in the network. We first describe the simulation setup and then compare the simulation results with GPSR [12], DWSIGF [13], QGOR [14], and our proposed FAGOR protocol in a variety of experiments. Next, we illustrate the advantage of the FA-UOFC over the traditional OFC approach without considering misbehavior of faulty nodes. Finally, we show the effectiveness of our proposed FAGOR protocol combined with our FA-UOFC algorithm for WSNs in adversarial environments, and we simulate the fairness of our proposed scheme in terms of utility fairness index and the convergence discussed in Section 6.3.1.

The extensive simulations have been conducted in OPNET and C++ simulator. The OPNET simulator is

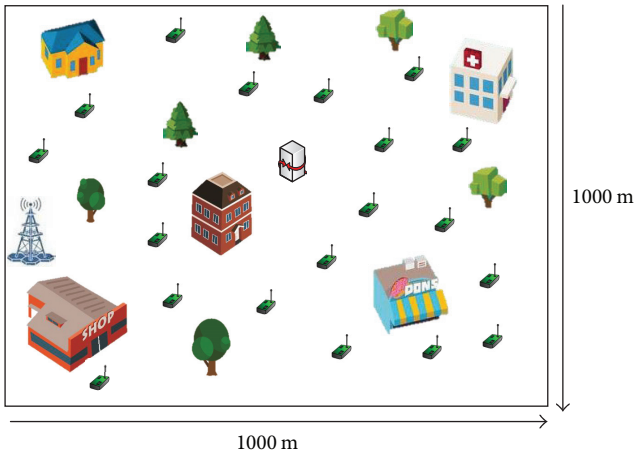


Figure 11: Simulation scenario.

designed for the network design and performance test. It is further enhanced to support for wireless sensor networks in city environments. In original OPNET, the calculation of received power only considers the propagation model of free space. In the urban communication environment, wireless channel is affected by the diffraction of signals by various buildings and trees. A Rician model is used as a channel fading model to illustrate effects due to buildings, obstacles, and trees in the city. We incorporate Rician distribution into the receiver power module in OPNET in accordance with radio wave propagation model in practical scenarios.

We consider static WSNs for a smart city. Therefore, mobility is not considered in experiments. As shown in Figure 11, 100 to 400 wireless sensors, which include both misbehaving sensors and well-behaved sensors, are randomly deployed in an area of 1000 m × 1000 m. The percentage of misbehaving nodes to all the nodes which is a simulation parameter is varied from 0 to 0.4 in different experiments. Each sensor has IEEE 802.15.4 based technology. The sources send data to 10 sinks which have sufficient power. The initial power of each sensor is set to 9 mW. The parameters for energy consumption are set to $e^t = 150$ nJ/bit, $e^r = 158$ nJ/bit, and $e^s = 100$ nJ/bit, respectively [50]. Each simulation runs 3000 iterations, and the default simulation parameters are listed in Table 1.

7.1. The Effectiveness of FAGOR. In this section, we show how our FAGOR protocol can provide effective routing with the existence of an arbitrary number of misbehaving nodes. The proposed FAGOR protocol is benchmarked against other three routing protocols: (1) DWSIGF, (2) GPSR, and (3) QGOR (a QoS-aware GOR which provides routing service based on the end-to-end QoS metric [22]). The following two metrics are used to compare the performance of the protocols:

(i) PDR: the ratio of the total number of data packets by the sink packet delivery to the total amount of data packets sent by the source

(ii) End-to-end delay: the time interval for the data packet to be transmitted from the source node to the sink

Table 1: Parameter values in simulations.

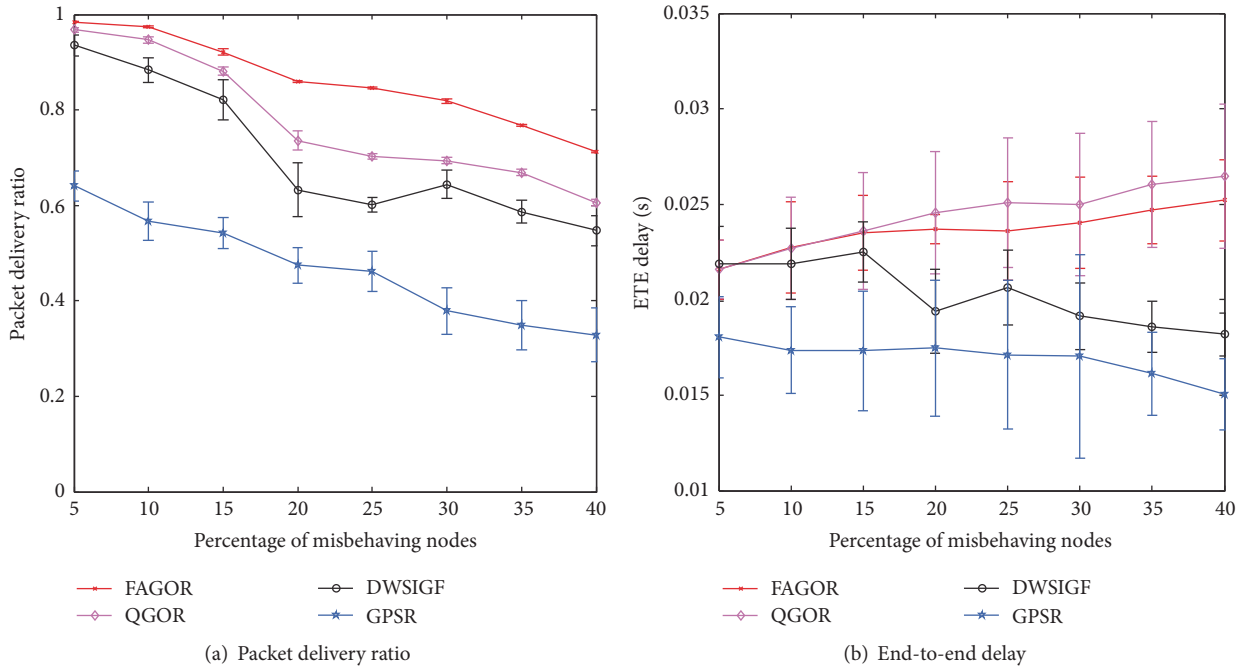
Parameter	Value
Simulation iterations	3000
Numbers of nodes	100, 200, 300 or 400
Percentage of misbehaving nodes	0~0.4
Network size	1000
MAC protocol	802.15.4
Packet size	512 byte
Numbers of candidates	$N = 3$
Maximum power consumption	9 mW
Power parameters	$e^t = 150$ nJ/bit, $e^r = 158$ nJ/bit, $e^s = 100$ nJ/bit
Weight values	$\alpha = 0.7$, $\delta_1 = 0.4$, $\delta_2 = 0.6$, $\kappa = 0.7$ $\gamma = 0.7$, $\vartheta = 0.7$, $\epsilon = 0.8$, $\varphi = 0.002$

We simulate Sybil attacks with 4 Sybil nodes which perform random attacks with a configurable probability. The Sybil nodes create more virtual locations by altering their transmission power, which is similar to location spoofing attackers. We model randomly distributed misbehavior nodes such as black holes, gray holes, and nodes in jamming regions which drop data packets with variable possibility. The routing protocol is simulated attacking with varied probabilities to evaluate performance under various misbehaviors.

First we show the effectiveness of FAGOR under varied the number of misbehaving nodes. Figure 12(a) reports the packet delivery ratio of FAGOR in comparison with the other three routing protocols. We have the following observations: (a) the PDR of FAGOR is consistently higher than GPSR and DWSIGF with the existence of a varied number of misbehaving nodes, and (b) the PDR of FAGOR declines more slowly than GPSR and DWSIGF as the percentage of misbehaving nodes increases. The reason is that the misbehaving nodes are more likely to be chosen as the next-hop nodes in GPSR and DWSIGF, while FAGOR incorporates faulty impacts for choosing more reliable candidates to set up the routing paths.

The PDR in QGOR is higher than in other routing protocols except FAGOR. This can be explained as follows. QGOR also selects more reliable relays according to the QoS priority of neighboring nodes. However, without the ability to identify location-related attacks, QGOR may select a Sybil node as the next-hop relay. Our FAGOR gives low reliability values to Sybil nodes based on majority voting and to other misbehaving nodes based on direct-impact values. In terms of the compound of reliability value by the proposed FA metric, FAGOR transmits packets with faulty hops, and the impact of misbehaviors on the network performance is stable.

As the number of misbehaving nodes increases, the end-to-end delay of GPSR and DWSIGF plotted in Figure 12(b) decreases. For hostile sensor networks, misbehaving nodes in the routing path would cause links to break. The decline of the end-to-end delay means that only the data packets



(a) Packet delivery ratio

(b) End-to-end delay

Figure 12: Packet delivery ratio and end-to-end delay versus percentage of misbehaving nodes.

from the nodes that are closer to the sink can be successfully delivered to the sink in GPSR and DWSIGF, while it is hard to successfully transmit the data packets to a distant destination with more hops. However, FAGOR and QGOR encourage suboptimal candidates to collaboratively relay data packets that the delay of such packets raises. As the number of misbehaving nodes increases, FAGOR and QGOR spend more time maintaining uninterrupted communication, and higher end-to-end delays are consequently achieved.

Furthermore, FAGOR gets a lower end-to-end delay than QGOR because of the existence of Sybil nodes among misbehaving nodes. Since the reliability of neighbors is unknown at the beginning, FAGOR uses majority voting to decrease the probability of location attacks. Compared to QGOR which operates without identifying location attacks, FAGOR mitigates Sybil attacks in advance and saves the network delay time.

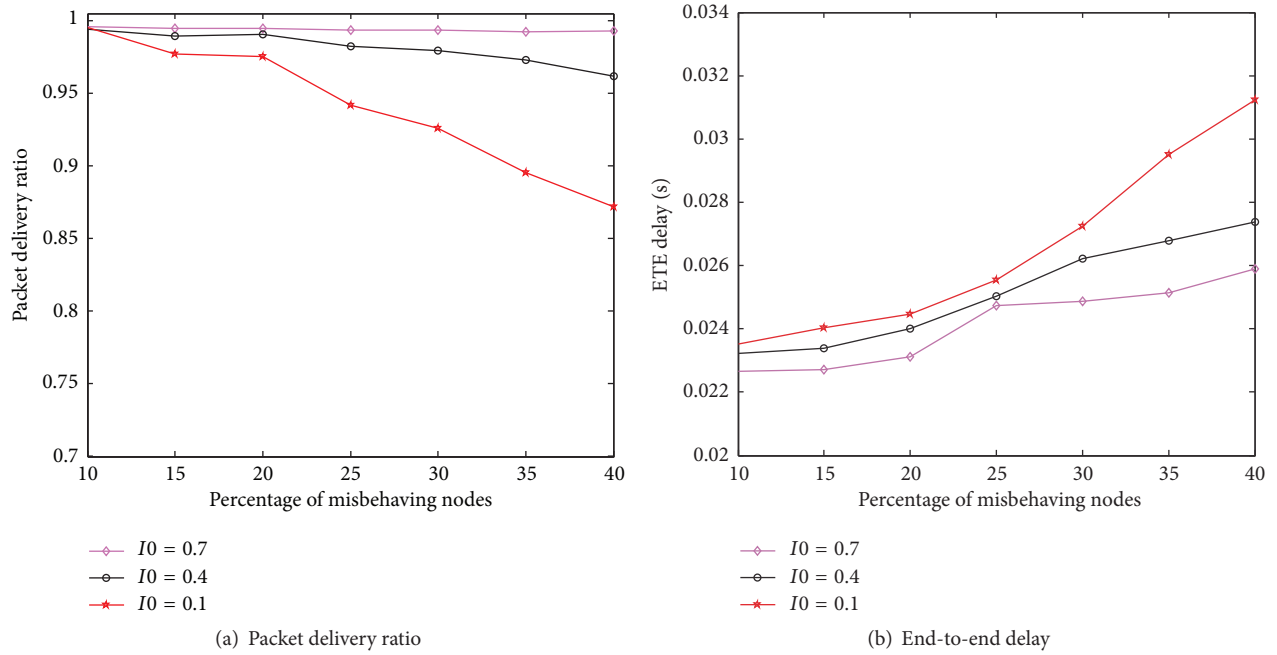
We further study the effect of I_0 on the performance of FAGOR. The packet delivery ratio under varied values of I_0 is shown in Figure 13(a). In this simulation, we find out that underestimating the parameter I_0 will lead to imprecise next-hop choosing results and will affect the performance of FAGOR. On the other hand, overestimating I_0 as shown in Figure 13(b) may make the routing algorithm yield less feasible next-hops, lead to repeated candidate discovery, and result in higher delay. This result illustrates that there is trade-off between the PDR and time delay and choosing a proper value of I_0 gives better performance of FAGOR.

Figure 14 compares the performance of four protocols for different network size by increasing the numbers of nodes from 100 to 400. Compared with GPSR and DWSIGF, our FAGOR improves the delivery ratio by approximately 40% and keeps stable with the different random topologies.

In order to evaluate the number of candidates of the performance of FAGOR, we consider network scenarios with different numbers of misbehaving nodes. From Figure 15(a), we see that PDR increases and the gap of PDR between $I_0 = 0.1$, $I_0 = 0.4$, and $I_0 = 0.7$ gets smaller as the number of candidates increases. Thus more candidates in FAGOR can relieve the performance degradation under more misbehaving nodes. Figure 15(b) shows that the transmission delay decreases when $N = 1$. This is because, in FAGOR, when packet dropping ratio is high, there will be fewer hop counts which means that the data delivery would not last long. As the number of candidates increases, transmission time delay when $I_0 = 0.1$ increases faster than when $I_0 > 0.1$ due to a long one-hop delay in the presence of more misbehaving nodes. The simulation results show that there is a trade-off between the time delay and robustness on the selection of the candidates' numbers.

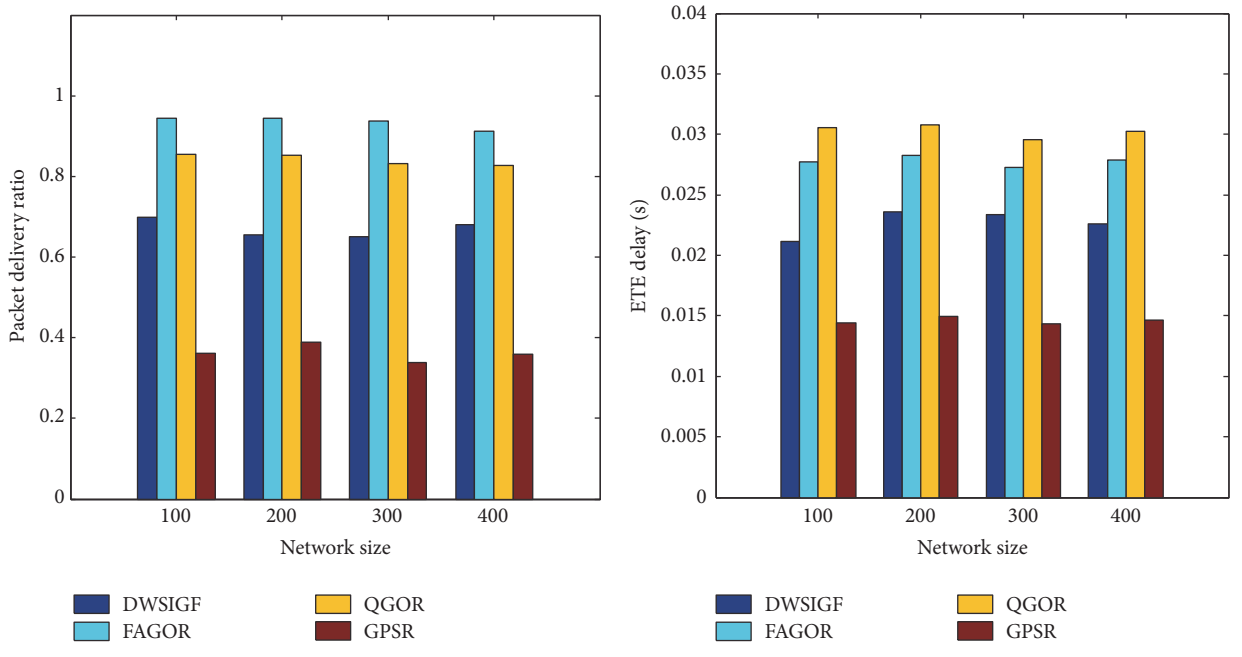
One object of FAGOR is to ensure the ability to operate effectively under dynamic misbehaving networks. In our simulation study, we set up a configurable probability of misbehaving nodes which behave well at the beginning of the experiment. They change to misbehaving nodes at random points of time. In Figure 16, we show the PDR performance of four protocols with a varied percentage of behavior-changing nodes. The following observations can be obtained from these figures. First, the packet delivery ratio of FAGOR is consistently higher than that of the other three protocols with different percentages of changing misbehaving nodes. Second, since FAGOR selects faulty nodes in the routing path, the impact of misbehaviors on the network performance is stable.

7.2. The Effectiveness of FA-UOFC. In this subsection, we use numerical examples to illustrate the advantage of FA-UOFC



(a) Packet delivery ratio (b) End-to-end delay

FIGURE 13: Packet delivery ratio and end-to-end delay with different values of I_0.

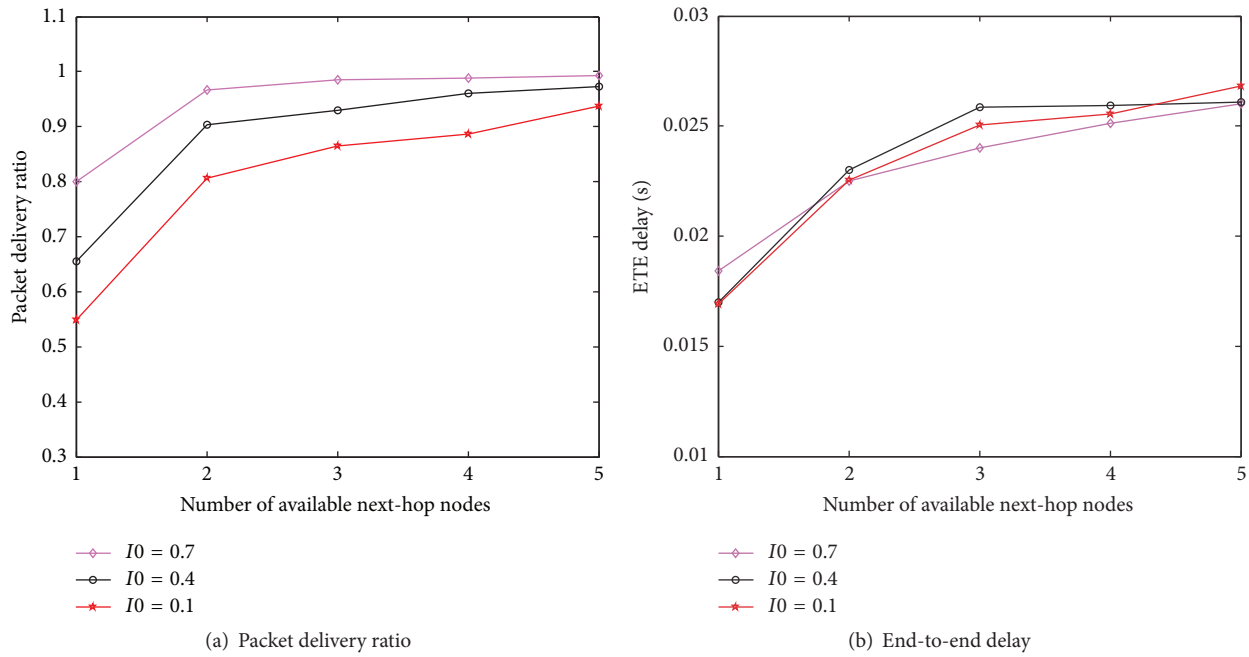


(a) Packet delivery ratio (b) End-to-end delay

FIGURE 14: Scalability evaluation.

algorithm over the OFC with same resource constraints. In the simulation, the sensor nodes turn to misbehaving nodes with probability 0.35. The network topology for one sink is depicted in Figure 17. We assume a link capacity of 4 kbps and a maximum node power consumption of 4 mW. In smart cities, there are various types of sensors embedded in networks to support multiple services with different QoS requirements. Therefore, we set utility functions consisting of elastic and inelastic traffic. The utility function of each source node is given as $U_1(x_1) = 1/(1 + e^{-2(x_1-6)})$, $U_2(x_2) = \log(x_2+1)/\log 11$, $U_3(x_3) = 0.1x_3$, $U_4(x_4) = 1/(1+e^{-2(x_4+4)})$. All the sources have their maximum rates at 10 Mbps.

We compare the effectiveness of two flow control strategies: (1) NE-OFC (OFC with noneffective utility



(a) Packet delivery ratio (b) End-to-end delay

Figure 15: Candidate number evaluation.

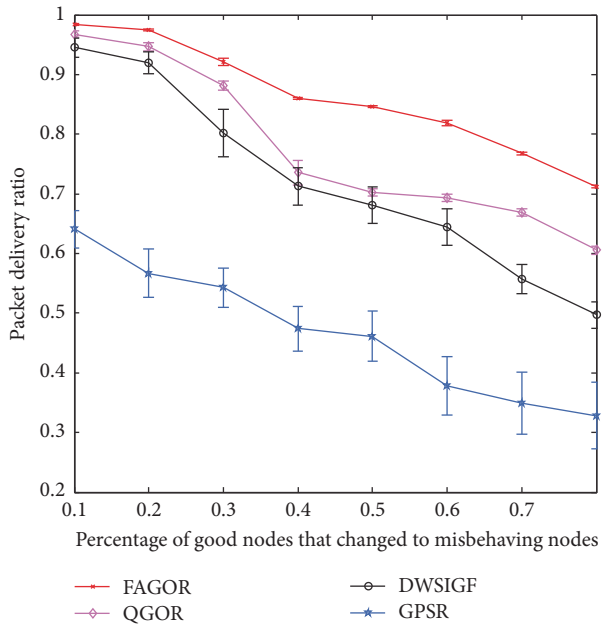


Figure 16: Packet delivery ratio versus percentage of behavior-changing nodes.

functions and constraints); (2) FA-UOFC (our improved OFC approach). NE-OFC approach subject to contention and energy constraints for WSNs is with utility functions of allocated flow rate without considering the faulty impact caused by misbehaving nodes. Figure 18 shows the comparison of the goodput for each flow at sink between our proposed FA-UOFC and NE-OFC. The proposed FA-UOFC can be seen to have achieved higher performance in terms of effective throughput compared to the conventional flow control method. Obviously this is due to the introduction of the faulty activity metric. The source adjusts its flow rate on its route adaptively to compensate for data loss in our FA-UOFC algorithm, which takes into account the effect of misbehaving nodes in utility function and constraints.

According to Section 6, x is denoted as the injection rate at the source node and x' is denoted as the goodput at the sink. Figure 19 verifies that the rate-control algorithm in NE-OFC converges and is able to provide utility proportional fairness (we use the sum of contention price and energy price) among four source nodes according to the utilities of x on the source nodes. Without considering faulty nodes, the source algorithm controls the flow rates to provide a utility fair resource allocation in which S_1 achieves a utility $U_1(x_1) = 1$ and S_2, S_3, and S_4 then share the remaining network resources with an equal utility of 0.52.

In fact, the goodputs of four flows cannot maintain the utility fairness at their sink nodes after traveling along the leaky-hops. The utilities of goodputs for four flows in the NE-OFC approach and FA-UOFC approach are shown in Figure 20. It can be seen that FA-UOFC yields higher utilities of goodput for four flows than NE-OFC. In Figure 19, three flows share a fair utility allocation that $U_2(x_2)$ is equal to $U_3(x_3)$ and $U_4(x_4)$. However, the utility fairness is broken due to different faulty effects on three paths consisting of misbehaving nodes. $U_3(x_3')$ and $U_4(x_4')$ of goodputs at the sinks both from NE-OFC and FA-UOFC in Figure 20 are lower than those of rates at the source nodes in Figure 19. Meanwhile, $U_2(x_2')$ of goodput from FA-UOFC increases, yet $U_2(x_2')$ from NE-OFC decreases. We calculate two indexes of utility fairness, 0.7 and 0.86, according to (29) for NE-OFC and FA-OFC, respectively. It demonstrates that better

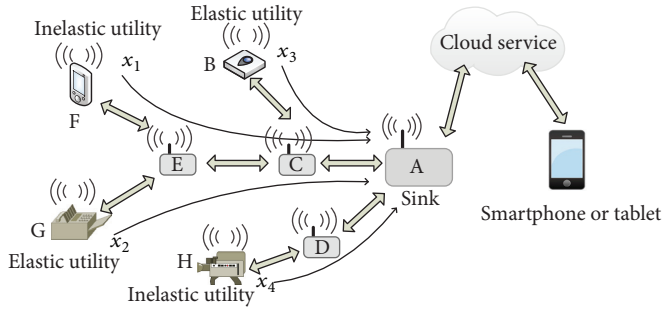


Figure 17: The network topology for one sink.

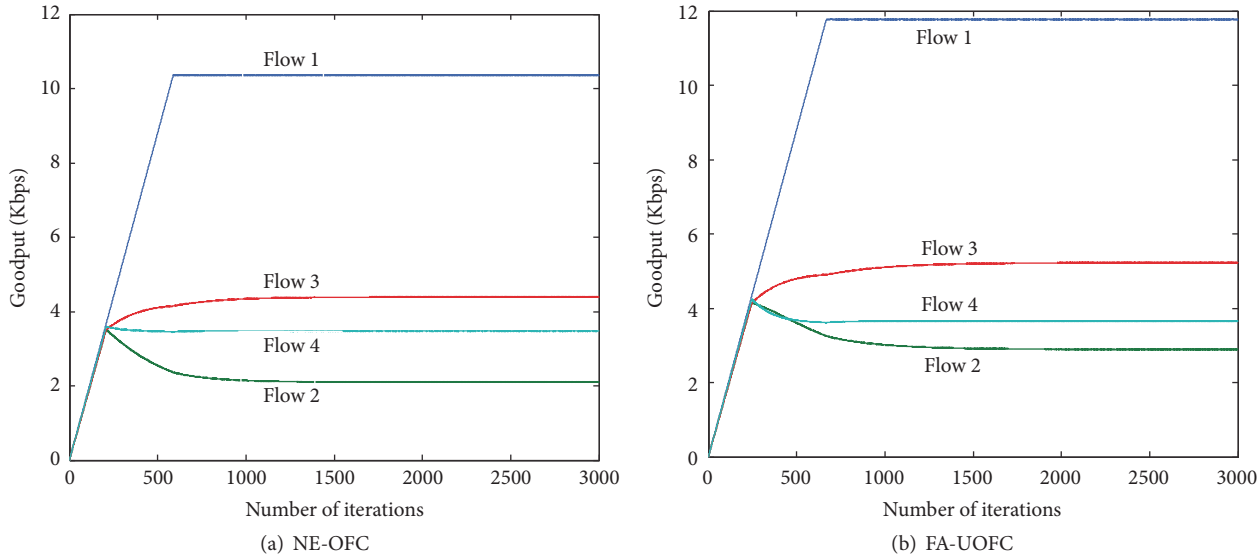


Figure 18: Goodput at sink.

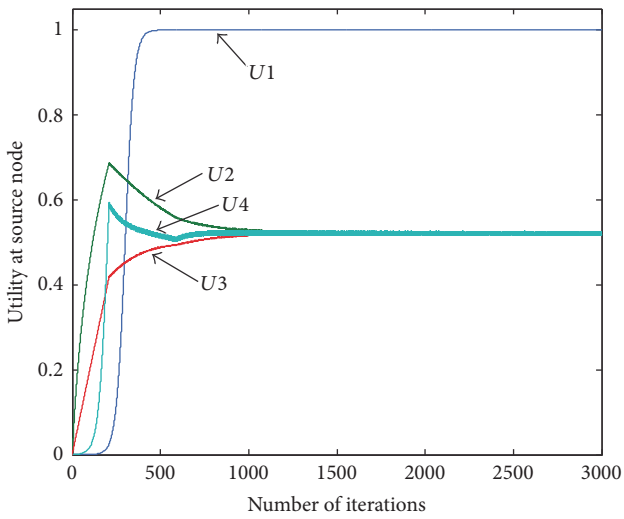
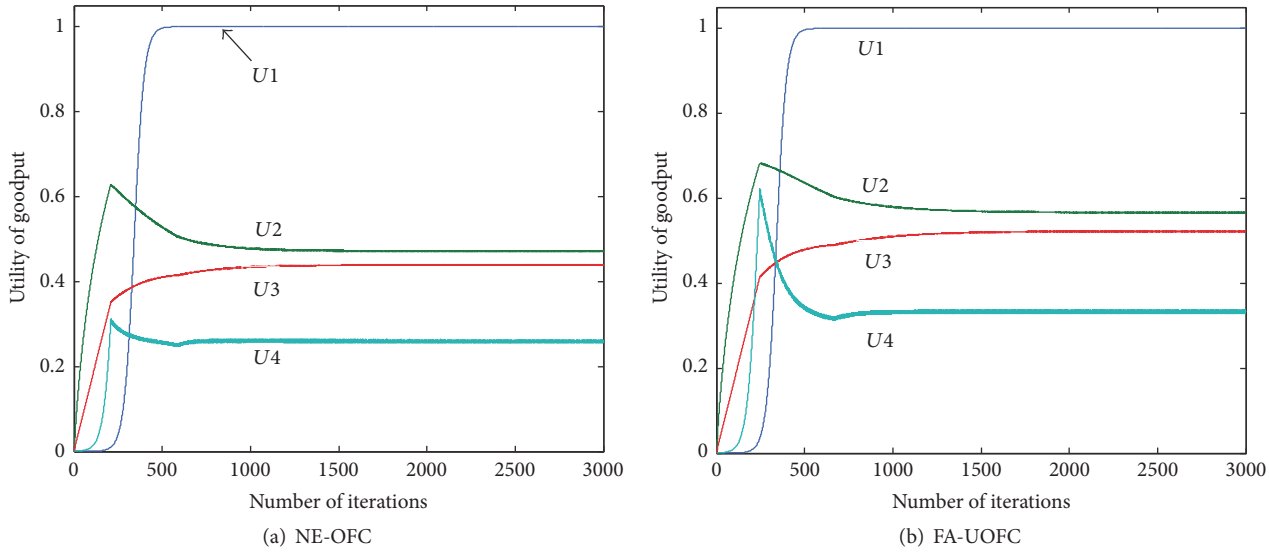


Figure 19: Utility of flow rate at source in NE-OFC.

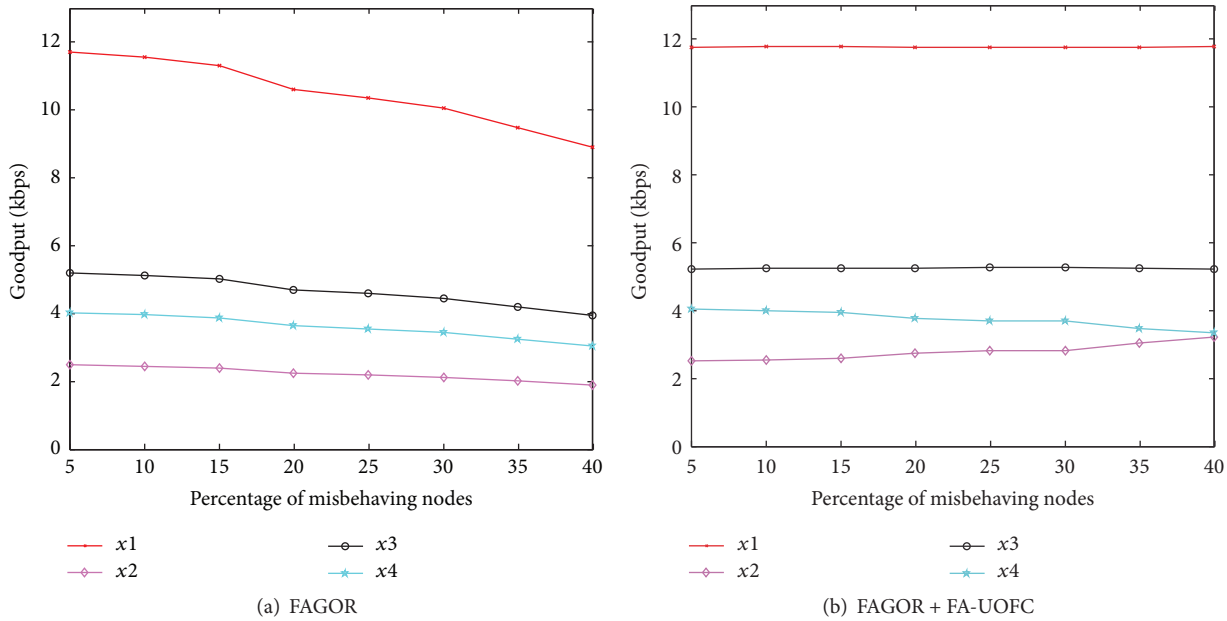
utility fairness is attained among flows by FA-UOFC. Our proposed algorithm effectively adjusts the resource allocation by explicitly taking into account the faulty effects in utility functions and constraints. Clearly, the network performance under misbehaving nodes is improved by our proposed FA-UOFC algorithm through both better utility fairness and higher effective throughput.

7.3. The FAGOR Protocol Combined with FA-UOFC Algorithm. In the following, we investigate the performance of our proposed FAGOR protocol combined with FA-UOFC algorithm for WSNs in adversarial environments. The proposed FAGOR + FA-UOFC scheme is benchmarked against the scheme with only FAGOR which does not employ any optimal flow control algorithm. Figures 21 and 22 plot the goodputs and the goodputs' utilities obtained by FAGOR and FAGOR + FA-UOFC while increasing the percentage of misbehaving nodes in the network from 5% to 40%. Clearly, our proposed method significantly outperforms FAGOR in terms of the goodputs and goodputs' utilities obtainable under a varied percentage of misbehaving nodes. The benefit of our proposed method over FAGOR increases as the number of misbehaving nodes increases. The result demonstrates that the FA-UOFC complements secure routing and alleviates the performance degradation caused by the misbehaving nodes along the routing paths.



(a) NE-OFC (b) FA-UOFC

Figure 20: Utility of goodput.



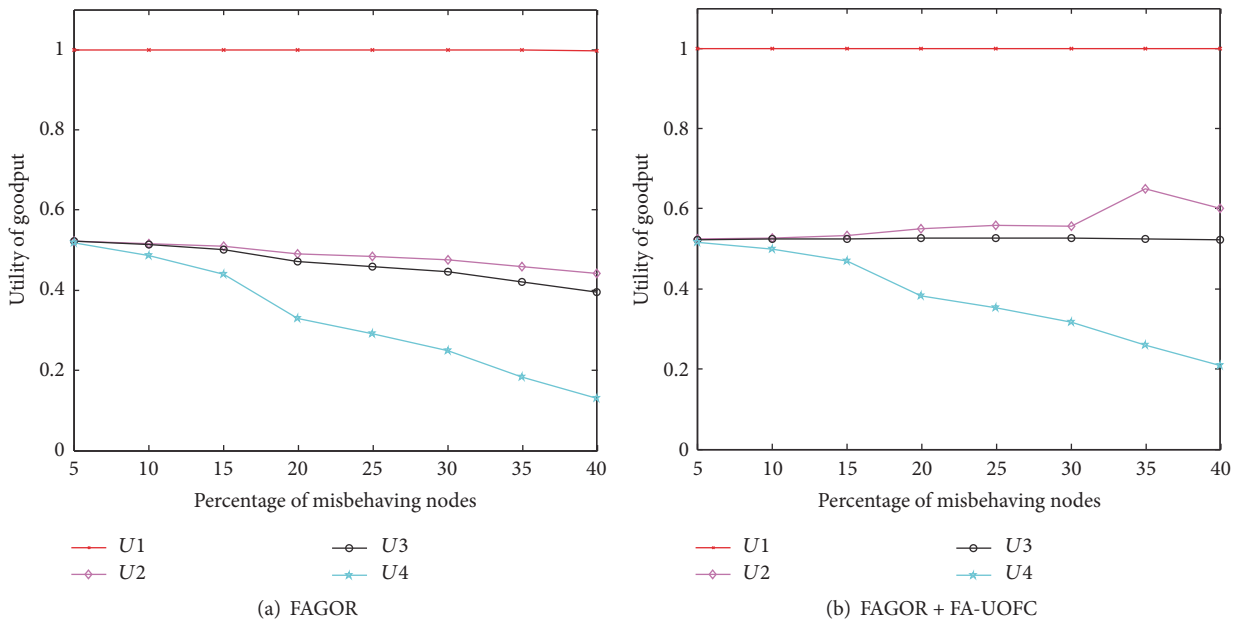
(a) FAGOR (b) FAGOR + FA-UOFC

Figure 21: Goodput at sink.

We also take a closer look at Flow 2 and Flow 3 in Figure 22. As the number of the misbehaving nodes increases, the goodputs' utilities of Flow 2 and Flow 3 in our scheme increase, whereas they decrease in FAGOR. Accordingly, our scheme achieves higher goodputs' utilities for Flow 2 and Flow 3 than FAGOR. This is due to the source nodes in our scheme, which are able to compensate for faulty nodes in the allocation of traffic based on the real performance requirements of services and which can achieve utility fairness among the goodputs.

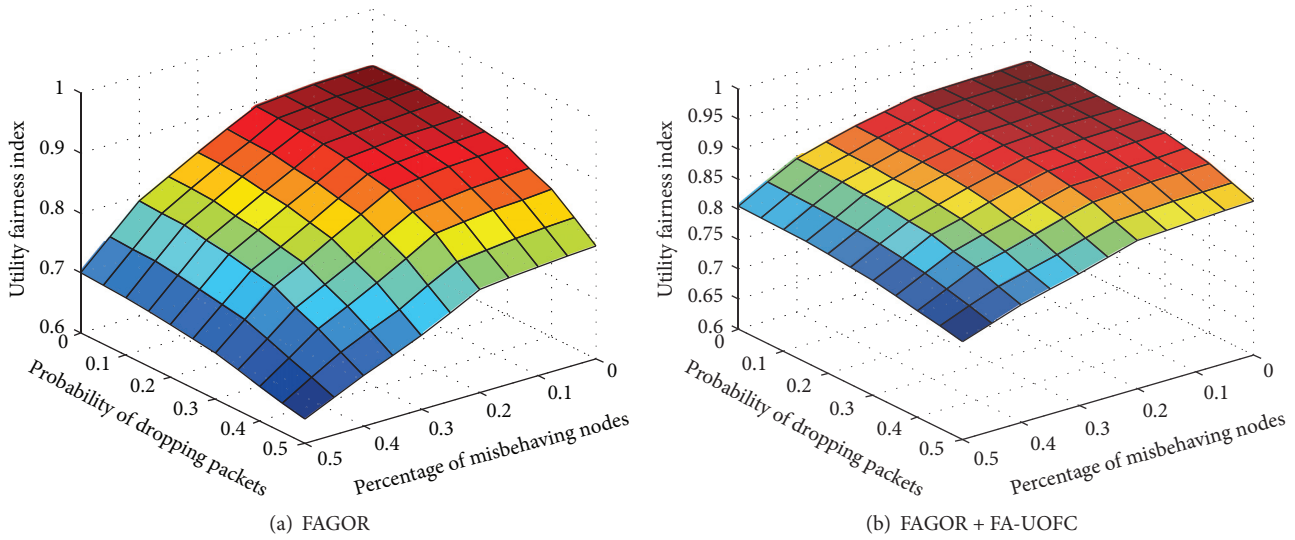
To demonstrate the fairness of FAGOR and FAGOR + FA-UOFC, we point to the variation of $f(x)$ in (29). With various values for the percentage of misbehaving nodes p_1 and the probability of dropping packets p_2 in Figure 23, our proposed scheme can be seen to achieve a higher degree of utility fairness in terms of utility fairness index $f(x)$ for goodput than the FAGOR scheme. This is because our proposed scheme explicitly takes into account the loss feature of faulty nodes and embodies the utility fairness objectives in the utility function that are concerned with the goodputs.

For a sequence of networks with decreasing impact with misbehaving nodes, we can see in Figure 23 that the utility fairness index converges to 0.92. As discussed in Section 6, the rate allocation and utility fairness in our scheme converge to those of the corresponding lossless networks when the ratios of nodes' faulty activities drop to zero. Figure 23



(a) FAGOR (b) FAGOR + FA-UOFC

Figure 22: Utility of each flow.



(a) FAGOR (b) FAGOR + FA-UOFC

Figure 23: Utility fairness index.

shows the trends of utility fairness for goodput in adversarial environments.

8. Conclusion

In this paper, we studied the problem of routing and rate control for multiple city services over wireless sensor networks in the presence of misbehaving nodes whose effect can be characterized statistically. We presented methods for each sensor to probabilistically characterize the impact of a variable fault. To address how to maintain an acceptable level of network performance degradation, we utilized fault activity information in the next-hop selection of each sensor and incorporated this information into the rate-control algorithm for data sources. An improved, fault-aware version of the routing algorithm FAGOR is proposed, and we explicitly added fault activity information into the routing metric. We formulated resource allocation for multiple services as a lossy network flow optimization problem using relaxed utility functions. In addition, we developed a distributed rate-control algorithm called FA-UOFC which can achieve the lossy utility fairness among sources with different traffic types. Through comprehensive performance comparisons, we demonstrate that FAGOR protocol achieves a better performance with an acceptable overhead and that FA-UOFC algorithm achieves a higher effective utility and better utility

fairness when various misbehaving nodes exist in a WSN. Finally, we show that our proposed FAGOR protocol combined with FA-UOFC algorithm proves effective in improving effective utility and utility fairness compared to the scheme with only FAGOR protocol.

Even through the development of our research is based on the wireless sensor network setting, the framework can generally be extended to other energy-constrained wireless ad hoc network models. In the future, mobility aspects can be considered in order to model more realistic wireless networks in smart cities. We also plan to model smart malicious behaviors and study their effects on data delivery.

Conflicts of Interest

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grants nos. 61373154, 61672239, and 61632012) and Shanghai High Technology Field Project (Grant no. 16511101400).

References

[1] F. Sanchez-Rosario, D. Sanchez-Rodriguez, J. B. Alonso-Hernandez et al., "A low consumption real time environmental monitoring system for smart cities based on ZigBee wireless sensor network," in *Proceedings of the 11th International Wireless Communications and Mobile Computing Conference, IWCMC 2015*, pp. 702–707, August 2015.

[2] M. Díaz, C. Martín, and B. Rubio, "State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing," *Journal of Network and Computer Applications*, vol. 67, pp. 99–117, 2016.

[3] J. Xu, K. Wang, C. Wang et al., "Byzantine fault-tolerant routing for large-scale wireless sensor networks based on fast ECDSA," *Tsinghua Science and Technology*, vol. 20, no. 6, Article ID 7350015, pp. 627–633, 2015.

[4] B. Zhang, Z. H. Huang, and Y. Xiang, "A novel multiple-level trust management framework for wireless sensor networks," *Computer Networks*, vol. 72, pp. 45–61, 2014.

[5] I. M. Atakli, H. Hu, Y. Chen, W. Ku, and Z. Su, "Malicious node detection in wireless sensor networks using weighted trust evaluation," in *Proceedings of the Spring Simulation Multiconference (SpringSim '08)*, pp. 836–843, April 2008.

[6] W. Wang, M. Chatterjee, K. Kwiat, and Q. Li, "A game theoretic approach to detect and co-exist with malicious nodes in wireless networks," *Computer Networks*, vol. 71, pp. 63–83, 2014.

[7] J. Hao, R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan, "Sparse Malicious False Data Injection Attacks and Defense Mechanisms in Smart Grids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1198–1209, 2015.

[8] C. M. Ahmed, S. Adepu, and A. Mathur, "Limitations of state estimation based cyber attack detection schemes in industrial control systems," in *Proceedings of the 2016 Smart City Security and Privacy Workshop, SCSP-W 2016*, pp. 6–10.

[9] J. Wu, K. Ota, M. Dong, and C. Li, "A hierarchical security framework for defending against sophisticated attacks on wireless sensor networks in smart cities," *IEEE Access*, vol. 4, pp. 416–424, 2016.

[10] A. S. Lal and R. Nair, "Region authority based collaborative scheme to detect Sybil attacks in VANET," in *Proceedings of the International Conference on Control, Communication and Computing India, ICCC 2015*, pp. 664–668, November 2015.

[11] K. Zeng, J. Yang, and W. Lou, "On energy efficiency of geographic opportunistic routing in lossy multihop wireless networks," *Wireless Networks*, vol. 18, no. 8, pp. 967–983, 2012.

[12] B. Karp and H. T. Kung, "GPSR: greedy Perimeter Stateless Routing for wireless networks," in *Proceedings of the MobiCom*, pp. 243–254, Boston, Mass, USA, 2000.

[13] I. A. Umar, Z. M. Hanapi, A. Sali, and Z. A. Zulkarnain, "A forwarding strategy for DWSIGF routing protocol," in *Proceedings of the 5th International Conference on IT Convergence and Security, ICITCS 2015*, August 2015.

[14] L. Cheng, J. Niu, J. Cao, S. K. Das, and Y. Gu, "QoS aware geographic opportunistic routing in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1864–1875, 2014.

[15] J. M. Gormally and R. L. Richards, "Application layer protocols for disruption tolerant remote sensor SATCOM links," in *Proceedings of the 33rd Annual IEEE Military Communications Conference, MILCOM 2014*, pp. 975–982, October 2014.

[16] I. Al-Anbagi, M. Erol-Kantarci, and H. T. Mouftah, "A survey on cross-layer quality-of-service approaches in WSNs for delay and reliability-aware applications," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 1, pp. 525–552, 2016.

[17] G. Hosseinabadi and N. Vaidya, "Selfish misbehavior in scheduling algorithms of wireless networks," in *Proceedings of the 2010 IEEE 29th International Performance Computing and Communications Conference, IPCCC 2010*, pp. 214–221, December 2010.

[18] M. Tahir and S. K. Mazumder, "Delay constrained optimal resource utilization of wireless networks for distributed control systems," *IEEE Communications Letters*, vol. 12, no. 4, pp. 289–291, 2008.

[19] Y. Li, M. Chiang, A. R. Calderbank, and S. N. Diggavi, "Optimal rate-reliability-delay tradeoff in networks with composite links," in *Proceedings of the IEEE INFOCOM 2007: 26th IEEE International Conference on Computer Communications*, pp. 526–534, May 2007.

[20] W. H. Wang, M. Palaniswami, and S. H. Low, "Application-oriented flow control: fundamentals, algorithms and fairness," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1282–1291, 2006.

[21] M. Chaqfeh, N. Mohamed, I. Jawhar, and J. Wu, "Vehicular cloud data collection for Intelligent Transportation Systems," in *Proceedings of the 3rd Smart Cloud Networks and Systems, SCNS 2016*, December 2016.

[22] J. Tang, W. P. Tay, and T. Q. S. Quek, "Cross-layer resource allocation with elastic service scaling in cloud radio access network," *IEEE Transactions on Wireless Communications*, vol. 14, no. 9, pp. 5068–5081, 2015.

[23] P. Spachos, D. Toumpakaris, and D. Hatzinakos, "QoS and energy-aware dynamic routing in wireless multimedia sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '15)*, pp. 6935–6940, London, UK, June 2015.

[24] J. Luo, D. Wu, C. Pan, and J. Zha, "Optimal Energy Strategy for Node Selection and Data Relay in WSN-based IoT," *Mobile Networks and Applications*, vol. 20, no. 2, pp. 169–180, 2015.

[25] X. Kang and Y. Wu, "Incentive Mechanism Design for Heterogeneous Peer-to-Peer Networks: A StackelbergGame Approach," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 1018–1030, 2015.

[26] I.-R. Chen, F. Bao, M. Chang, and J.-H. Cho, "Dynamic trust management for delay tolerant networks and its application to secure routing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 5, pp. 1200–1210, 2014.

[27] R. Mitchell and I.-R. Chen, "Effect of intrusion detection and response on reliability of cyber physical systems," *IEEE Transactions on Reliability*, vol. 62, no. 1, pp. 199–210, 2013.

[28] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attacks and network defense policies in wireless sensor networks," in *Proceedings of the IEEE INFOCOM 2007: 26th IEEE International Conference on Computer Communications*, pp. 1307–1315, May 2007.

[29] W. Wang, M. Chatterjee, and K. Kwiat, "Coexistence with malicious nodes: A game theoretic approach," in *Proceedings of the 2009 International Conference on Game Theory for Networks, GameNets '09*, pp. 277–286, May 2009.

[30] R. Mitchell and I.-R. Chen, "Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 16–30, 2015.

[31] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, 2003.

[32] D. Wu, J. Luo, R. Li, and A. Regan, "Geographic load balancing routing in hybrid vehicular ad hoc networks," in *Proceedings of the 14th IEEE International Intelligent Transportation Systems Conference (ITSC '11)*, pp. 2057–2062, IEEE, Washington, DC, USA, October 2011.

[33] K. Liu, N. Abu-Ghazaleh, and K.-D. Kang, "Location verification and trust management for resilient geographic routing," *Journal of Parallel and Distributed Computing*, vol. 67, no. 2, pp. 215–228, 2007.

[34] F. P. Kelly, A. K. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 206–217, 1997.

[35] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: a mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.

[36] Y. Xue, L. I. Baochun, and K. Nahrstedt, "Optimal resource allocation in wireless ad hoc networks: a price-based approach," *IEEE Transactions on Mobile Computing*, vol. 5, no. 4, pp. 347–364, 2006.

[37] S. Eswaran, A. Misra, F. Bergamaschi, and T. La Porta, "Utility-based bandwidth adaptation in mission-oriented wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 8, no. 2, article no. 17, 2012.

[38] J.-W. Lee, R. R. Mazumdar, and N. B. Shroff, "Non-convex optimization and rate control for multi-class services in the internet," *IEEE/ACM Transactions on Networking*, vol. 13, no. 4, pp. 827–840, 2005.

[39] P. Hande, S. Zhang, and M. Chiang, "Distributed rate allocation for inelastic flows," *IEEE/ACM Transactions on Networking*, vol. 15, no. 6, pp. 1240–1253, 2007.

[40] A. Boukerche, H. H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Secure localization algorithms for wireless sensor networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 96–101, 2008.

[41] L. Tan, X. Zhang, L. L. H. Andrew, S. Chan, and M. Zukerman, "Price-based max-min fair rate allocation in wireless multi-hop networks," *IEEE Communications Letters*, vol. 10, no. 1, pp. 31–33, 2006.

[42] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, "Four-bit wireless link estimation," in *Proceedings of the HotNets VI*, 2007.

[43] K. Zeng, W. Lou, J. Yang, and D. R. Brown III, "On throughput efficiency of geographic opportunistic routing in multihop wireless networks," *Mobile Networks and Applications*, vol. 12, no. 5-6, pp. 347–357, 2007.

[44] K. Zeng, K. Govindan, D. Wu, and P. Mohapatra, "Identity-based attack detection in mobile wireless networks," in *Proceedings of the IEEE INFOCOM 2011*, pp. 1880–1888, April 2011.

[45] M. Demirbas and Y. Song, "An RSSI-based scheme for sybil attack detection in wireless sensor networks," in *Proceedings of the WoWMoM 2006: 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pp. 564–568, June 2006.

[46] S. Yang, C. K. Yeo, and B.-S. Lee, "Toward reliable data delivery for highly dynamic mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 1, pp. 111–124, 2012.

[47] V. G. Subramanian, K. R. Duffy, and D. J. Leith, "Existence and uniqueness of fair rate allocations in lossy wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, pp. 3401–3406, 2009.

[48] Z. Cao and E. W. Zegura, "Utility max-min: an application-oriented bandwidth allocation scheme," in *Proceedings of the IEEE 18th Annual Joint Conference of Computer and Communications Societies (INFOCOM '99)*, vol. 2, pp. 793–801, New York, NY, USA, March 1999.

[49] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Tech. Rep., 1984.

[50] "Chipcon Inc, CC2420, True single-chio 2.4 GHz IEEE 802.15.4/ZigBee RF transceiver with MAC support," http://www.chipcon.com.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 4687974, 9 pages
https://doi.org/10.1155/2017/4687974

Research Article

Crowdsensing Task Assignment Based on Particle Swarm Optimization in Cognitive Radio Networks

Linbo Zhai[1,2] and Hua Wang[2]

[1]*Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Shandong Normal University, Jinan, China*
[2]*School of Computer Science and Technology, Shandong University, Jinan, China*

Correspondence should be addressed to Linbo Zhai; zhai@mail.sdu.edu.cn

Received 26 April 2017; Revised 18 July 2017; Accepted 2 August 2017; Published 11 September 2017

Academic Editor: Bin Guo

Cognitive radio technology allows unlicensed users to utilize licensed wireless spectrum if the wireless spectrum is unused by licensed users. Therefore, spectrum sensing should be carried out before unlicensed users access the wireless spectrum. Since mobile terminals such as smartphones are more and more intelligent, they can sense the wireless spectrum. The method that spectrum sensing task is assigned to mobile intelligent terminals is called crowdsourcing. For a large-scale region, we propose the crowdsourcing paradigm to assign mobile users the spectrum sensing task. The sensing task assignment is influenced by some factors including remaining energy, locations, and costs of mobile terminals. Considering these constraints, we design a precise sensing effect function with a local constraint and aim to maximize this sensing effect to address crowdsensing task assignment. The problem of crowdsensing task assignment is difficult to solve since we prove that it is NP-hard. We design an optimal algorithm based on particle swarm optimization to solve this problem. Simulation results show our algorithm achieves higher performance than the other algorithms.

1. Introduction

In recent years, the wireless traffic has grown heavily and this case leads to crowd wireless spectrum. According to the current policy that wireless spectrum assignment is fixed, only licensed users can utilize the licensed wireless spectrum. Even though the wireless spectrum is idle, unlicensed users cannot use the idle spectrum. Therefore, the current policy of spectrum assignment leads to low ratio of wireless spectrum utilization. To solve this problem, cognitive radio has recently emerged to improve wireless spectrum utilization [1]. When the licensed wireless spectrum is idle, cognitive radio makes unlicensed users utilize the wireless spectrum. Therefore, unlicensed users should carry out spectrum sensing before they use the wireless spectrum.

With the development of mobile terminals such as smartphones and pads, a new paradigm called mobile crowd sensing and computing (MCSC) appears [2]. The formal definition of MCSC is described as follows: a new sensing paradigm that empowers ordinary citizens to contribute data sensed or generated from their mobile devices and aggregates and fuses the data in the cloud for crowd intelligence extraction and human-centric service delivery.

Inspired by MCSC, mobile terminals configured with sensors are leveraged to accomplish spectrum sensing task. In the same spirit, with the recent Federal Communications Commission (FCC) ruling that a geolocation database could be used by Secondary TV spectrum users to obtain the spectrum availability, it is assumed that there is a crowdsourcing-based fusion center (FC). FC assigns sensing task to mobile users and receives the sensing data from them. To incentivize mobile users to carry out sensing tasks, FC needs to provide monetary benefits. This way is called crowdsourcing.

In this paper, we propose the crowdsourcing paradigm to assign the spectrum sensing task to many mobile users. It is assumed that there is a crowdsourcing-based fusion center (FC). FC assigns the sensing task to mobile users. During the assignment process, we have considered some factors.

At first, the remaining energy is very important to mobile users. Only when a mobile user has enough energy can the wireless spectrum be sensed. Then mobile users should be given incentives to carry out spectrum sensing. With a limited budget, FC may choose a subset of whole mobile users to carry out spectrum sensing. At last, the positions of mobile users also influence the sensing results. Considering these factors, we propose precise sensing effect function for the crowdsourcing-based sensing task assignment. And the objective function considers a local constraint. Then we prove that the sensing task assignment is NP-hard. Therefore, we design an optimal algorithm based on particle swarm optimization (PSO) to solve the problem. Simulation results show our proposed algorithm achieves higher performance than other algorithms.

In this paper, we study the problem of sensing task assignment. The main contributions of this paper are summarized below.

(i) Considering the remaining energy of mobile users, budget constraint, and mobile users' positions, we propose precise objective function with a local constraint. We define the local constraint which means the sensing effect of a channel in a location is not less than a threshold. Compared to other literatures, we aim to not only maximize global sensing effect but also satisfy the local sensing constraint. And we prove the sensing task assignment is NP-hard.

(ii) Since the sensing task assignment is NP-hard, we design an optimal algorithm based on particle swarm optimization (PSO) to solve the problem. To the best of our knowledge, there is no related work designing the PSO-based algorithm to solve sensing task assignment in cognitive radio networks.

(iii) Simulation results show our proposed algorithm achieves higher performance than other algorithms.

The rest of the paper is organized as follows. In Section 2, related literatures are introduced. In Section 3, the system model of sensing task assignment is described. In Section 4, we design a PSO-based algorithm to solve the sensing task assignment. In Section 5, the proposed algorithm is evaluated with simulation results. Finally, conclusions are shown in Section 6.

2. Related Work

In cognitive radio networks, licensed users activity will decide whether the spectrum is idle or not [3]. As some factors such as shadowing and multipath fading may make a user mistake the sensing result, cooperative spectrum sensing is proposed to improve the sensing accuracy [4].

There have been some related literatures about cooperative spectrum sensing. In wideband wireless system, users exchange their compressed sensing results. According to the sensing results, they estimate the spectrum states cooperatively [5]. In [6], authors propose a two-level defense scheme to solve the attackers in cooperative spectrum sensing. In [7], cooperative spectrum sensing based on crowdsourcing is studied to address the security issue brought by malicious mobile users. In [8], authors consider the simultaneous sensing and transmitting of users and propose a novel detection model for cooperative spectrum sensing. In multichannel networks, the sensing task assignment is considered in parallel, and several sensing strategies are proposed to schedule users based on network parameters [9]. In [10], authors propose a game-theoretic distributed power control mechanism based on channel sensing results of users in cognitive wireless sensor network. To maximize the sensing quality, authors study the problem of multichannel sensing assignment in the multichannel system [11–13]. These literatures use a simplistic objective function and there is no budget constraint. If the system has a limited budget, there may be only a subset of mobile users chosen to carry out spectrum sensing. In [14], considering budget constraint, the authors study the problem of sensing task and channel allocation. However, the energy of mobile users is not considered. In [15], considering the character of sensing tasks and the sensor availability, authors study the multitask allocation problem to maximize overall system utility. It is the first to study different data quality metrics and formulate the multitask allocation optimization problem when diverse sensing capability constraints of each participant are taken into account. To achieve the near-optimal objective, the method using a two-phase offline multitask allocation framework needs historical call data from the telecom operator.

The aforementioned literatures use centralized algorithms. There are some distributed methods about spectrum sensing. In [16], with a distributed way, spatial spectrum sensing is studied to make use of spatial spectrum opportunities. To analyze the performance of spatial spectrum sensing, stochastic geometry is utilized. In [10], based on channel sensing results of users, a game-theoretic distributed power control mechanism is proposed. Besides, there are other studies about spectrum sensing [17–20].

3. System Model

It is assumed that there is a crowdsourcing-based fusion center (FC). FC assigns the sensing task to mobile users. Remaining energy and positions of mobile users, as well as limited budget, may influence the assignment process. Considering these constraints, we propose precise sensing effect function with a local constraint. Then we prove the sensing task assignment is NP-hard.

3.1. Problem Formulation. We assume that there are many locations needed to be sensed. In each location, there are many channels that needed sensing. By crowdsensing task assignment, we aim to maximize the sensing effect with a local constraint.

Let M denote the number of locations needed to be sensed and $N(j)$ denote the number of channels that should be sensed in a location j. In the location j, shadowing, multipath fading, and other issues may influence the sensing results of mobile users in different positions of this location. In other words, mobile users may obtain different sensing results in the same location since they are at different

positions. Therefore, location j may be divided into several sublocations. The spatial diversity can be captured by the sensing outcomes of mobile users in different sublocations. In a sublocation h of location j, $z^i_{hj} = 1$ denotes that there is at least one mobile user sensing channel i, and $z^i_{hj} = 0$ denotes that there are no mobile users sensing channel i. In location j, let y^i_j denote the number of sublocations where channel i is sensed by at least one mobile user. We can derive $y^i_j = \sum_{h=1}^{m(j)} z^i_{hj}$, where $m(j)$ denotes the number of sublocations in location j. Obviously, the higher y^i_j is, the more effective the sensing result is. When y^i_j equals zero, there is no sensing effect. When y^i_j equals $m(j)$, the maximized sensing effect is reached. We can imagine that sensing effect increases fast as y^i_j increases when y^i_j is small, while sensing effect increases slowly as y^i_j increases when y^i_j is large. Let $f(i,j) = \sqrt{y^i_j / m(j)}$ denote the sensing effect of channel i in location j. Then we can design the sensing effect function for the crowdsensing task assignment as follows:

$$\sum_{j=1}^{M} \sum_{i=1}^{N(j)} w^i_j f(i,j), \tag{1}$$

where w^i_j denotes the nonnegative weight with $\sum_{j=1}^{M} \sum_{i=1}^{N(j)} w^i_j = 1$, and w^i_j could distinguish the important degrees of sensing channels in each location. According to formula (1), the sensing effect function increases as y^i_j varies from zero to $m(j)$. And the smaller y^i_j is, the faster sensing effect function increases with the y^i_j growth. The larger y^i_j is, the more slowly sensing effect function increases with the y^i_j growth.

To obtain optimized sensing effect, we aim to maximize the sensing effect function in (1) with a local constraint which means the sensing effect of channel i in location j is no less than a threshold H. The local constraint can be described as

$$f(i,j) \geq H \quad i \in [1, N(j)], \ j \in [1, M]. \tag{2}$$

There are some factors which should be considered as follows.

For the mobile users, the remaining energy should be considered at first. Only when one mobile user's remaining energy is higher than the threshold could the mobile user carry out the task of spectrum sensing. Let Th be the normalized threshold of the remaining energy, $\mathbf{K}$ denote the set of all mobile users, and e_k be the remaining energy for a mobile user k. Then the energy constraint can be expressed as

$$e_k \geq \text{Th} \quad k \in \mathbf{K}. \tag{3}$$

Let M denote the number of locations needed to be sensed. For a location j, only the mobile users in that location can sense the channels within that location. We assume a mobile user can only sense one channel. In location j, let $\mathbf{K}(j)$ denote the set of mobile users, $n(j)$ denote the number of mobile users, and $N(j)$ denote the number of channels that should be sensed. For the mobile user $k \in \mathbf{K}(j)$, $x_{ki} = 1$ denotes that the channel i is sensed by mobile user k and $x_{ki} = 0$ denotes that the channel i is not sensed by mobile user k. Then considering a mobile user can only sense one channel, another constraint can be expressed as

$$\sum_{k \in \mathbf{K}(j)} \sum_{i=1}^{N(j)} x_{ki} \leq n(j). \tag{4}$$

Additionally, the incentive scheme allows FC to pay for the mobile users that try to sense channels. However, the cost of crowdsensing must be in the acceptable range. Let C be the maximum cost that can be paid for the sensing users and c_k be the cost for the mobile user $k \in \mathbf{K}(j)$. The constraint can be expressed as

$$\sum_{j=1}^{M} \sum_{k \in \mathbf{K}(j)} c_k \sum_{i=1}^{N(j)} x_{ki} \leq C. \tag{5}$$

The optimal object of crowdsensing task assignment can be described as

$$\begin{aligned}
\max \quad & \sum_{j=1}^{M} \sum_{i=1}^{N(j)} w^i_j f(i,j) \\
\text{subject to} \quad & f(i,j) \geq H, \quad i \in [1, N(j)], \ j \in [1, M] \\
& e_k \geq \text{Th}, \quad k \in \mathbf{K} \\
& \sum_{k \in \mathbf{K}(j)} \sum_{i=1}^{N(j)} x_{ki} \leq n(j) \\
& \sum_{j=1}^{M} \sum_{k \in \mathbf{K}(j)} c_k \sum_{i=1}^{N(j)} x_{ki} \leq C \\
& \sum_{j=1}^{M} \sum_{i=1}^{N(j)} w^i_j = 1.
\end{aligned} \tag{6}$$

Figure 1 depicts an example of crowdsensing task assignment. There are two locations and three channels in the system. Each location is divided into three sublocations. Mobile users in different sublocations may obtain different sensing results about the same channel. Since the local constraint is not satisfied or the remaining energy is not enough or the cost is too high, some mobile users are not assigned sensing task. Other users are assigned channels to sense according to formula (6).

3.2. NP-Hardness. The problem of crowdsensing task assignment is difficult to solve since we prove this problem is NP-hard. The reason is that the problem of crowdsensing task assignment is as hard as maximum coverage problem which is NP-hard [21].

The maximum coverage problem is described as follows: given a number d and a collection of l sets $S = \{S_1, S_2, \ldots, S_l\}$,

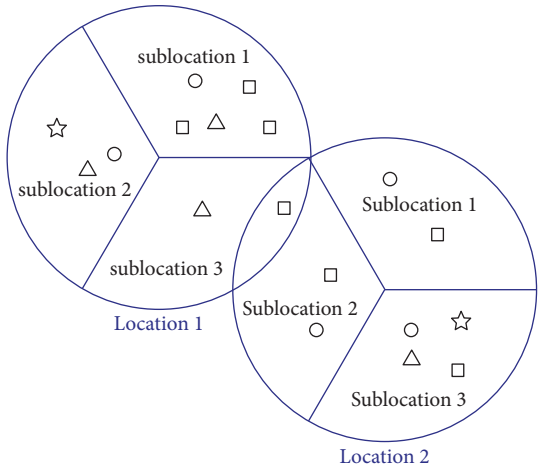


Figure 1: An example of crowdsensing task assignment.

the maximum coverage problem is to select at most d of these sets to form S' such that the maximum number of elements is covered:

$$\max_{S'} \quad \left| \bigcup_{S_i \in S'} S_i \right|, \qquad \text{s.t.} \quad S' \subseteq S, \quad \left|S'\right| \leq d. \tag{7}$$

Theorem 1. *The problem of crowdsensing task assignment is NP-hard.*

Proof. By showing a special case of crowdsensing task assignment is as hard as maximum coverage problem, we prove that the problem of crowdsensing task assignment is NP-hard.

The special case is described as follows: each mobile user has enough energy to carry out spectrum sensing, the local threshold H is set to zero that means the local constraint is satisfied, and the cost of crowdsensing is in the acceptable range. There are K mobile users and M locations in the system. And, in each location, there are N channels that should be sensed. Each mobile user is denoted by $i \in \{1, 2, \ldots, K\}$. Then K mobile users can form 2^K sets such as $\{1\}$, $\{1, 2\}$, and $\{1, 2, 3\}$. Let the nonnegative weight $w^i{}_j$ be a constant. Then (6) can be rewritten as

$$\max \sum_{j=1}^{M} \sum_{i=1}^{N} f(i, j). \tag{8}$$

Let l equal 2^K and d equal MN. Equation (8) means selecting d sets from l sets to maximize the sum of $f(i, j)$. Compared to (7), it is at least as hard as the maximum coverage problem which is NP-hard. In other words, the special case of crowdsensing task assignment is NP-hard.

The problem of crowdsensing task assignment is no easier than the special case. Therefore, the problem of crowdsensing task assignment is NP-hard. □

4. The Optimal Algorithm Based on PSO

Since the crowdsensing task assignment problem is NP-hard, we design the optimal algorithm based on particle swarm optimization (PSO) to solve this problem in this section. The PSO algorithm is good at NP-hard problem optimization [22]. The PSO algorithm is described at first. Then the optimal algorithm based on PSO is proposed. And time complexity is analyzed.

4.1. PSO Algorithm. In the PSO algorithm [23], each particle flies in the search space with certain speed. During the flight, a particle changes its flight experience with its companions. Therefore, each particle can fly to a better solution region based on this mechanism. Let V_{id} denote the particle speed and X_{id} denote the particle's position. The movement of the particle is described as follows:

$$V_{id}^{t+1} = wV_{id}^{t} + c_1 r_1 \left(P_{id} - X_{id}^{t}\right) + c_2 r_2 \left(P_{gd} - X_{id}^{t}\right) \tag{9}$$

$$X_{id}^{t+1} = V_{id}^{t+1} + X_{id}^{t}, \tag{10}$$

where w denotes the inertia weight, P_{id} denotes this particle's historical best position, and P_{gd} denotes the global best position. Both r_1 and r_2 are independent in the range $[0, 1]$, and both c_1 and c_2 are study factors. The inertia weight w makes the algorithm improve its performance according to a series of applications. Formulas (9) and (10) calculate the current particle's velocity and position, respectively.

4.2. Crowdsensing Task Assignment Algorithm Based on PSO. We design an optimal algorithm based on PSO to solve crowdsensing task assignment. According to PSO algorithm, each particle's position represents a solution to the crowdsensing task assignment problem. It can be denoted by a matrix as follows.

When there are $N(j)$ channels in location $j \in [1, M]$, the total number of sensing channels is $\sum_{j=1}^{M} N(j)$ in all locations. Let K denote the number of mobile users. Then each particle is defined as a $K \times \sum_{j=1}^{M} N(j)$ matrix $\mathbf{X}$, where $\mathbf{X}[a][b] = 1$ denotes that the mobile user a chooses channel b to sense, and $\mathbf{X}[a][b] = 0$ denotes that the mobile user a does not choose channel b to sense.

We optimize the crowdsensing task assignment based on PSO algorithm (PSO-CTA). The optimized algorithm is described as follows. Initialize q particles randomly, and each particle denotes a solution of crowdsensing task assignment of all K mobile users. Then we set the particle with the highest objective function based on formulas (6) to be the current best solution. According to the PSO algorithm, we use the PSO formulas (9) to merge the crowdsensing task assignment and determine the new particle position until it converges or this swarm obtains its longest lifetime. If PSO-CTA converges, the best solution can be obtained. The proposed algorithm is described as follows.

Initialization. The first important problem to be solved is how the algorithm initially produces the particles. We produce a random particle as follows.

For a mobile user, its remaining energy should be consid ered at first. If its remaining energy is higher than the threshold, the mobile user could carry out the task of spectrum sensing. Then it chooses a channel to sense randomly in its corresponding locations. All mobile users with enough energy choose channels like this. If the local constraint of sensing effect in (2) is satisfied in each location, the cost should be considered next. Otherwise, this particle should be generated again. If the cost for the mobile users which carry out spectrum sensing is lower than the maximum cost C, the process of initialization is completed. If the cost for the mobile users which carry out spectrum sensing is higher than the maximum cost C, FC will not assign sensing task to some mobile users to satisfy the cost constraint. At first, when there are multiple users sensing a channel in the same sublocation, FC will only choose a user with lower cost to assign sensing task, and other users are given up. According to our model, the sensing effect will not change. If the cost constraint is satisfied, the initialization is completed. Otherwise, FC should continue to give up users in the sublocations with less weight until the cost constraint is satisfied. Then a $K \times \sum_{j=1}^{M} N(j)$ matrix $\mathbf{X}$ is generated corresponding to this particle.

Initialize q particles randomly, and each particle denotes a solution of crowdsensing task assignment of all K mobile users.

Optimizing Process. After each spectrum sensing instance of a mobile user, its energy will decrease. A mobile user should determine that its remaining energy meets the energy constraint. If its remaining energy is higher than the threshold, the mobile user is able to carry out spectrum sensing again. If its remaining energy is lower than the threshold, the mobile user could not carry out spectrum sensing from now on. For each particle, if a mobile user's energy is not enough to carry out spectrum sensing, the user's row vector is set to zero in the corresponding matrix. Then the matrix of a particle will change.

Based on the current matrix, the crowdsensing effect function of the particle is obtained following (1). After calculating all particles' effect function, we can derive a particle's historical best position P_{id} and the global best position P_{gd}. The best position corresponds to the maximized crowdsensing effect function.

According to a particle's historical best position P_{id} and the global best position P_{gd}, we merge the matrixes to optimize the sensing task assignment. Let $\mathbf{T_1}$ denote the current matrix of a particle and $\mathbf{T_2}$ and $\mathbf{T_3}$ denote historical best solution of the particle and the global best solution, respectively. The merging matrix can be described as the combination of $\mathbf{T_1}$, $\mathbf{T_2}$, and $\mathbf{T_3}$. Then we optimized the merging matrix as follows.

In the merging matrix, if a channel in a sublocation is sensed by multiple users, only one user with higher energy is reserved and other users are given up. That means only an element is set to one in the column vector of the merging matrix after optimization. If a user chooses different channels to sense in $\mathbf{T_1}$, $\mathbf{T_2}$, and $\mathbf{T_3}$, there are more than elements set to one in the row vector of the merging matrix. Considering the global property of PSO, we optimize the row vectors of the merging matrix with specific probability decided by the parameters in (9) to guarantee the search space. If a mobile user chooses different channels in these three matrixes, the user will select the channel in $\mathbf{T_1}$ based on the probability $w/(w + c_1 + c_2)$, select the channel in $\mathbf{T_2}$ based on the probability $c_1/(w + c_1 + c_2)$, and select the channel in $\mathbf{T_3}$ based on the probability $c_2/(w + c_1 + c_2)$. That means only an element is set to one in the row vector of the merging matrix after optimization. The search space and converging speed of this algorithm can be adjusted by adjusting the values of w, c_1, and c_2.

The proposed algorithm for crowdsensing task assignment problem is described in Algorithms 1, 2, and 3.

4.3. Analysis of Time Complexity. The complexity of proposed PSO-CTA algorithm is computed as follows. The computation complexity is $O(n \times N \times q)$ in the initialization stage, where n denotes the number of mobile users, N denotes the number of channels, and q denotes the number of particles.

In Line (3) of Algorithm 1, optimizing the sensing task assignment which is described in Algorithm 3 dominates the complexity of our algorithm. Then we focus on the computation complexity of optimizing the sensing task assignment. In a particle, the mobile users satisfying formulas (2), (3), (4), and (5) should be chosen, and the chosen mobile users are combined to obtain the maximized sensing effect function. Therefore, the complexity of a particle is $O(n^2 \times N)$. The complexity of all particles is $O(n^2 \times N \times q)$ at the stage of evaluating sensing effect function.

When particles update their velocities and positions, the computation complexity is $O(n \times q)$ in Lines (4)–(7) of Algorithm 1. Therefore, the computation complexity of the whole algorithm is $O(n^2 \times N \times q)$.

5. Simulation Results

The proposed PSO-CTA algorithm is evaluated by simulations. The average solution is obtained by running the algorithm 100 times. We compare our PSO-CTA algorithm with the algorithm in [14]. The simulation parameters are described as follows. There are some locations needed to be sensed, with the same radius. Each location is equally divided into 3 sublocations. The whole number of channels is $N = 5$. The local threshold H is set to 0.57. The nonnegative weight of $w^i{}_j$ is identical for each channel and each location. Mobile users are deployed randomly in the locations.

Figure 2 shows the crowdsensing effect outcomes as the number of locations varies from 15 to 40 when there are 50 mobile users. The cost values of c_k are chosen from $\{1; 2; 3; \ldots; 49; 50\}$. The maximum cost is $C = a \sum_{k=1}^{50} c_k$, where a equals 0.6 and 0.8, respectively. The normalized energy threshold of Th is set to 0.2 and 0.5, respectively. The crowdsensing effect function could be obtained based

Input:
Objective function according to formula (6);
A local constraint H;
The number of mobile users K;
The number of locations M;
The number of channels $N(j)$ in location j;
The number of sub-locations $m(j)$ in location j;
The maximum cost C;
The maximal generation T;
Output: The maximum sensing effect function and sensing task assignment
Initialization:
Randomly generate each particle;
Optimization:
(1) **repeat**
(2) **for** each particle
(3) Optimizing the crowdsensing task assignment of the particle;
(4) Update the P_{id};
(5) Update the P_{gd};
(6) **end for**
(7) **until** stopping criterion is satisfied

Algorithm 1: Overall procedure of proposed PSO-CTA.

(1) **for** each mobile user
(2) **if** its remaining energy satisfies formula (3)
(3) It chooses a random channel;
(4) **else** it will not sense;
(5) **end if**
(6) **end for**
(7) **if** the cost constraint is satisfied
(8) The initialization is completed;
(9) **else** reserve a user sensing a same channel in a sub-location;
(10) **end if**
(11) **if** the cost constraint is satisfied
(12) The initialization is completed;
(13) **else** give up users less weight until the cost is satisfied;
(14) **end if**

Algorithm 2: Random generation of each particle (initialization).

(1) **for** each particle
(2) **for** each mobile user
(3) **if** its remaining energy satisfies formula (3)
(4) Maintain the matrix;
(5) **else** set the corresponding row vector to zero;
(6) **end if**
(7) **end for**
(8) Then the current matrix $\mathbf{T_1}$ is derived;
(9) Evaluate the crowdsensing effect function of this particle;
(10) Obtainthis particle's P_{id} (matrix $\mathbf{T_2}$) based on crowdsensing effect function;
(11) Obtain P_{gd} (matrix $\mathbf{T_3}$) with the optimal P_{id};
(12) Merge matrix $\mathbf{T_1}$, $\mathbf{T_2}$, $\mathbf{T_3}$;
(13) Optimize the column vectors of the merging matrix;
(14) Optimize row vectors of the merging matrix with specific probability using (9);
(15) Evaluate the crowdsensing effect function of merging matrix according to (6);
(16) **end for**

Algorithm 3: Procedure of optimizing the sensing task assignment of the particle.

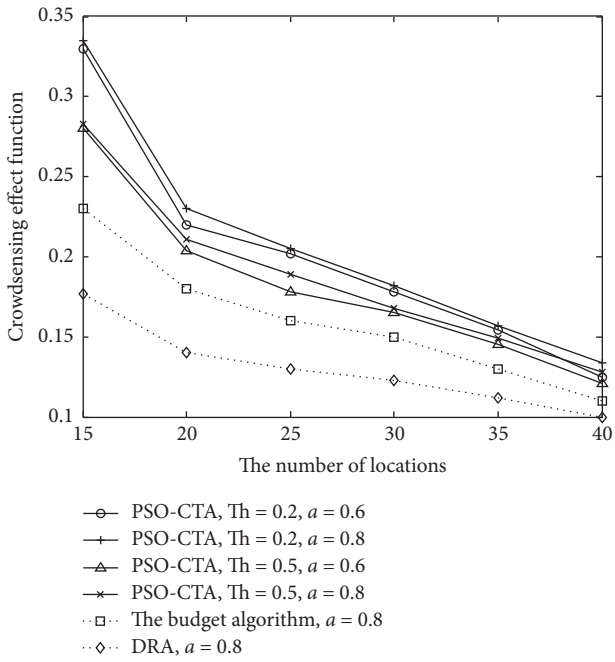


Figure 2: Crowdsensing effect function with 50 mobile users.

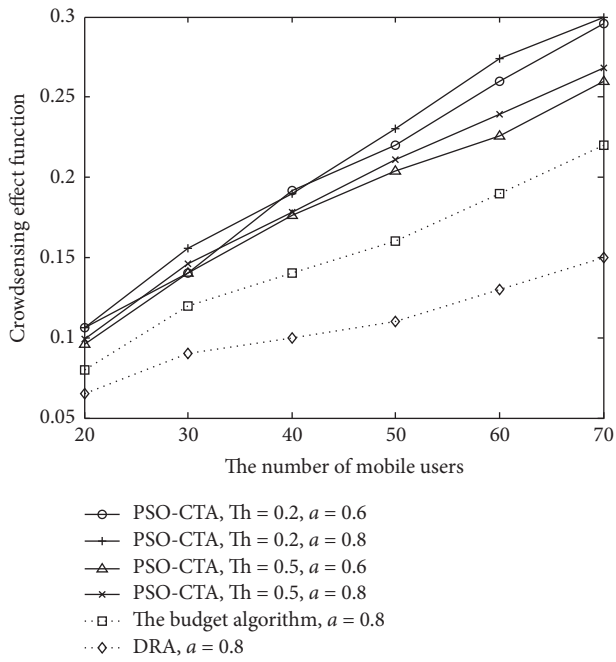


Figure 3: Crowdsensing effect function with 20 locations.

on (1). This function reflects the sensing effect and its value is between 0 and 1. The closer the value is to 1, the better the sensing effect is. Compared to the DRA algorithm in [13] and the budget algorithm in [14], our proposed PSO-CTA algorithm achieves higher crowdsensing effect function. As the number of locations increases, the crowdsensing effect function decreases. The reason is that more locations lead to more sublocations and fixed number of mobile users cannot sense all sublocations. When Th equals 0.2 and a equals 0.8, the crowdsensing effect function obtained is higher than those obtained when Th and a equal other values, since there are more mobile users assigned to sense channels with Th = 0.2 and $a = 0.8$.

Figure 3 shows the crowdsensing effect results as the number of mobile users varies from 20 to 70 when there are 20 locations. The cost values of c_k are chosen from $\{1; 2; 3; \ldots; L\}$, where L denotes the number of mobile users. The maximum cost is $C = a \sum_{k=1}^{L} c_k$, where a equals 0.6 and 0.8, respectively. The normalized energy threshold of Th is set to 0.2 and 0.5, respectively. Compared to the DRA algorithm in [13] and the budget algorithm in [14], our proposed PSO-CTA algorithm achieves higher crowdsensing effect function. As the number of mobile users increases, the crowdsensing effect function increases. The reason is that more sublocations could be sensed by more mobile users. When Th equals 0.2 and a equals 0.8, the crowdsensing effect function obtained is higher than those obtained when Th and a equal other values, since there are more mobile users assigned to sense channels with Th = 0.2 and $a = 0.8$.

Figure 4 shows the average remaining energy of mobile users as the number of spectrum sensing instances increases when there are 50 mobile users deployed randomly in 15 locations. It is assumed that the initial average energy of each user is 0.6. And after each spectrum sensing instance, a mobile user's energy falls 0.5%. The normalized energy threshold Th is set to 0.5 and 0.3, respectively. As shown in Figure 4, our proposed PSO-CTA algorithm achieves higher remaining energy of mobile users than the other algorithms. And we can see that the remaining energy will be higher when the threshold of Th is set to a higher value.

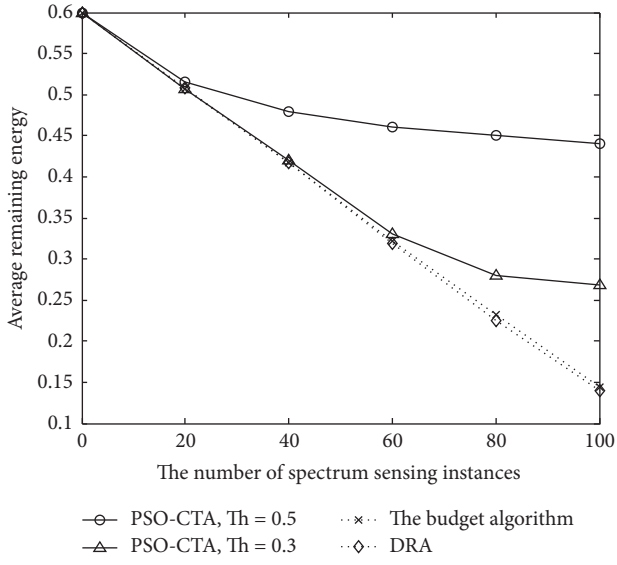


Figure 4: Average remaining energy with the number of spectrum sensing instances.

It is assumed that there are four channels and three locations which can be divided into three sublocations. The nonnegative weight is not identical for each channel. We

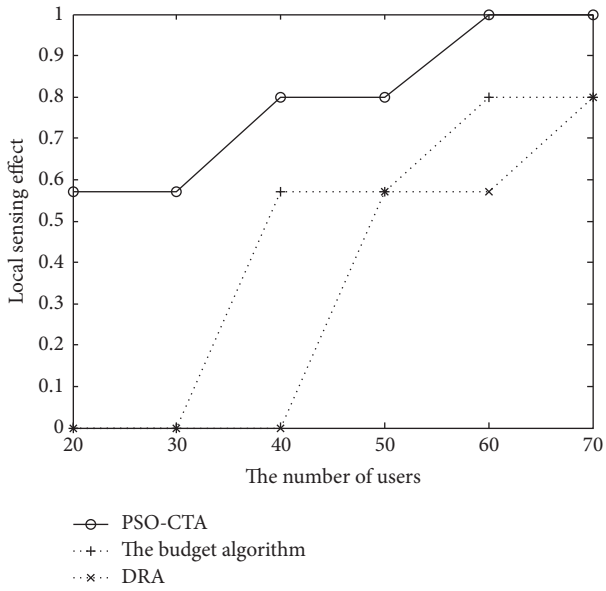


Figure 5: Local sensing effect with weight equaling 0.1 for three locations.

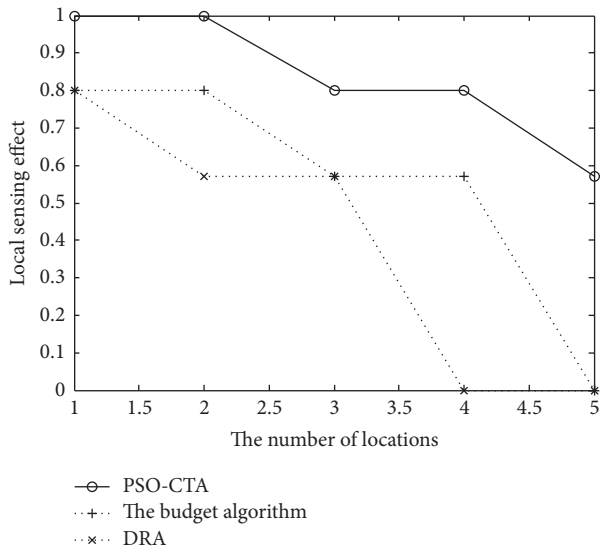


Figure 6: Local sensing effect with weight equaling 0.1 for 40 users.

set the weights equal to 0.3, 0.3, 0.3, and 0.1 for these four channels. Under the aforementioned conditions, the local sensing effect for the fourth channel (weight equaling 0.1) is shown in Figure 5. When there are not so many users in the system, the fourth channel is not sensed for the budget algorithm and DRA algorithm as the weight is too small to increase the global sensing effect. In the proposed PSO-CTA algorithm, the fourth channel should be sensed because a local constraint is set. Any channel, no matter what its weight equals, should be sensed. Therefore, no channel will be omitted with the PSO-CTA algorithm.

When there are 40 users, Figure 6 shows the local sensing effect for the fourth channel as the number of locations increases. There will not be enough users to sense each channel if the number of locations increases. Thus, the budget algorithm and DRA algorithm may choose the channels with higher weights to improve the global sensing effect. Therefore, the local sensing effect for the fourth channel will decrease. However, the proposed PSO-CTA algorithm will not ignore the fourth channel due to the local constraint.

6. Conclusion

For a large-scale region, this paper proposes the crowdsourcing method to assign the spectrum sensing task to many mobile users such as smartphones and pads. Considering some constraints such as remaining energy, locations, and costs of mobile users, we propose a sensing effect function with a local constraint and aim to maximize the sensing effect function. Since the problem of sensing task assignment is proved to be NP-hard, we design an optimal algorithm based on PSO to solve this problem. Simulation results show our algorithm achieves higher performance than the other algorithms.

Conflicts of Interest

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

Acknowledgments

This study is supported in part by National Natural Science Foundation of China (no. 61402270), Natural Science Foundation of Shandong Province, China (nos. BS2015DX003, ZR2014FQ009), Key Research and Development Program of Shandong Province, China (no. 2017GGX10142), and China Postdoctoral Science Foundation (no. 2014M561930).

References

[1] J. Mitola III and G. Q. Maguire Jr., "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.

[2] B. Guo, Z. Wang, Z. Yu et al., "Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm," *ACM Computing Surveys*, vol. 48, no. 1, article 7, 2015.

[3] Y. Saleem and M. H. Rehmani, "Primary radio user activity models for cognitive radio networks: a survey," *Journal of Network and Computer Applications*, vol. 43, pp. 1–16, 2014.

[4] T. Yücek 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.

[5] F. Zeng, Z. Tian, and C. Li, "Distributed compressive wideband spectrum sensing in cooperative multi-hop cognitive networks," in *Proceedings of the 2010 IEEE International Conference on Communications, ICC 2010*, zaf, May 2010.

[6] J. Feng, G. Lu, H. Wang, and X. Wang, "Supporting secure spectrum sensing data transmission against SSDH attack in cognitive radio ad hoc networks," *Journal of Network and Computer Applications*, vol. 72, pp. 140–149, 2016.

[7] R. Zhang, J. Zhang, Y. Zhang, and C. Zhang, "Secure crowdsourcing-based cooperative pectrum sensing," in *Proceedings of the 32nd IEEE Conference on Computer Communications, IEEE INFOCOM 2013*, pp. 2526–2534, April 2013.

[8] Y. Lu, D. Wang, and M. Fattouche, "Cooperative spectrum-sensing algorithm in cognitive radio by simultaneous sensing and BER measurements," *Eurasip Journal on Wireless Communications and Networking*, vol. 2016, no. 1, article no. 136, 2016.

[9] C.-H. Liu, A. Azarfar, J.-F. Frigon, B. Sansò, and D. Cabric, "Robust cooperative spectrum sensing scheduling optimization in multi-channel dynamic spectrum access networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 2094–2108, 2016.

[10] J. Zhu, D. Jiang, S. Ba, and Y. Zhang, "A game-theoretic power control mechanism based on hidden Markov model in cognitive wireless sensor network with imperfect information," *Neurocomputing*, vol. 220, pp. 76–83, 2017.

[11] P. Arora, N. Xia, and R. Zheng, "A Gibbs sampler approach for optimal distributed monitoring of multi-channel wireless networks," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference: "Energizing Global Communications", GLOBECOM 2011*, December 2011.

[12] D.-H. Shin, S. Bagchi, and C.-C. Wang, "Distributed online channel assignment toward optimal monitoring in multi-channel wireless networks," in *Proceedings of the IEEE Conference on Computer Communications, INFOCOM 2012*, pp. 2626–2630, March 2012.

[13] D.-H. Shin and S. Bagchi, "An optimization framework for monitoring multi-channel multi-radio wireless mesh networks," *Ad Hoc Networks*, vol. 11, no. 3, pp. 926–943, 2013.

[14] D.-H. Shin, S. He, and J. Zhang, "Joint sensing task and subband allocation for large-scale spectrum profiling," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 433–441, 2015.

[15] J. Wang, Y. Wang, D. Zhang, F. Wang, Y. He, and L. Ma, "Psaiiocator: Multi-task allocation for participatory sensing with sensing capability constraints," in *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing, CSCW 2017*, pp. 1139–1151, 2017.

[16] H. Chen, L. Liu, T. Novlan, J. D. Matyjas, B. L. Ng, and J. Zhang, "Spatial Spectrum Sensing-Based Device-to-Device Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 11, pp. 7299–7313, 2016.

[17] R. Sun, Y. Wang, X. Wang, and Y. Zhang, "Transceiver design for cooperative non-orthogonal multiple access systems with wireless energy transfer," *IET Communications*, vol. 10, no. 15, pp. 1947–1955, 2016.

[18] M. Jo, T. Maksymyuk, B. Strykhalyuk, and C.-H. Cho, "Device-to-device-based heterogeneous radio access network architecture for mobile cloud computing," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 50–58, 2015.

[19] X. Lin, J. G. Andrews, and A. Ghosh, "Spectrum sharing for device-to-device communication in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 12, pp. 6727–6740, 2014.

[20] Q. Liu, X. Wang, and Y. Cui, "Robust and adaptive scheduling of sequential periodic sensing for cognitive radios," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 3, pp. 503–515, 2014.

[21] D. S. Hochbaum, *Approximation Algorithm for NP-Hard Problems*, PWS Publishing Company, Massachusetts, 1997.

[22] J. Kennedy and R. Eberhart, "Particle swarm optimizatio," in *Proceedings of the IEEE International Conference on Neural Network*, pp. 1942–1948, Perth, Australia, 1995.

[23] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science (MHS '95)*, pp. 39–43, Nagoya, Japan, 1995.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 6834053, 16 pages
https://doi.org/10.1155/2017/6834053

Research Article

Data Dissemination Based on Fuzzy Logic and Network Coding in Vehicular Networks

Xiaolan Tang,[1] Zhi Geng,[1] Wenlong Chen,[1] and Mojtaba Moharrer[2]

[1]*College of Information Engineering, Capital Normal University, Beijing 100048, China*
[2]*Schepens Eye Research Institute, Massachusetts Eye and Ear, Harvard Medical School, Boston, MA 02114, USA*

Correspondence should be addressed to Wenlong Chen; chenwenlong@cnu.edu.cn

Received 31 March 2017; Revised 20 July 2017; Accepted 3 August 2017; Published 10 September 2017

Academic Editor: Petros Nicopolitidis

Vehicular networks, as a significant technology in intelligent transportation systems, improve the convenience, efficiency, and safety of driving in smart cities. However, because of the high velocity, the frequent topology change, and the limited bandwidth, it is difficult to efficiently propagate data in vehicular networks. This paper proposes a data dissemination scheme based on fuzzy logic and network coding for vehicular networks, named SFN. It uses fuzzy logic to compute a transmission ability for each vehicle by comprehensively considering the effects of three factors: the velocity change rate, the velocity optimization degree, and the channel quality. Then, two nodes with high abilities are selected as primary backbone and slave backbone in every road segment, which propagate data to other vehicles in this segment and forward them to the backbones in the next segment. The backbone network helps to increase the delivery ratio and avoid invalid transmissions. Additionally, network coding is utilized to reduce transmission overhead and accelerate data retransmission in interbackbone forwarding and intrasegment broadcasting. Experiments show that, compared with existing schemes, SFN has a high delivery ratio and a short dissemination delay, while the backbone network keeps high reliability.

1. Introduction

Vehicular networks, as a promising technology for intelligent transportation systems, usually utilize vehicle-to-vehicle (V2V) communications to support data services when no fixed infrastructures are deployed. They are designed to improve driving safety and enhance the driving experience by supporting smart applications such as collision warning, traffic congestion alarm, and sharing parking information [1–3]. When a collision occurs, this warning information should be immediately disseminated to those vehicles which might be affected, in order to avoid new rear-end collisions and potential traffic jams after this accident. An example scenario of safety alert dissemination is shown in Figure 1.

Data dissemination in vehicular networks faces many challenges including variation in vehicle densities, frequent topology change, and limited wireless communication bandwidth [4, 5]. In order to design an efficient data dissemination scheme aiming for high delivery ratio, short propagation delay, and low resource consumption, two aspects need to be considered: driving environments and content broadcasting. Complicated driving environments, as the first aspect of designing efficient data dissemination schemes, result in multiple factors affecting the intervehicle communication performance. Existing research analyzes some parameters in relay node selection, but the comprehensive influence of intervehicle distance, channel quality, and other factors still require further study. For content broadcasting, the second aspect of designing efficient data dissemination schemes, some schemes select different forwarding nodes for different data flows. Since wireless signals are likely to overlap with others in a geographical area, data dissemination in vehicular networks by flooding easily results in serious redundancy, contention, and collision [6]. With transmission demand increasing, the probability of broadcast storms may sharply rise. To address this issue, some studies select several nodes with excellent communication capabilities to disseminate data, which solve the broadcast storm problem in dense scenarios. However, there is still a lot of work to be done on

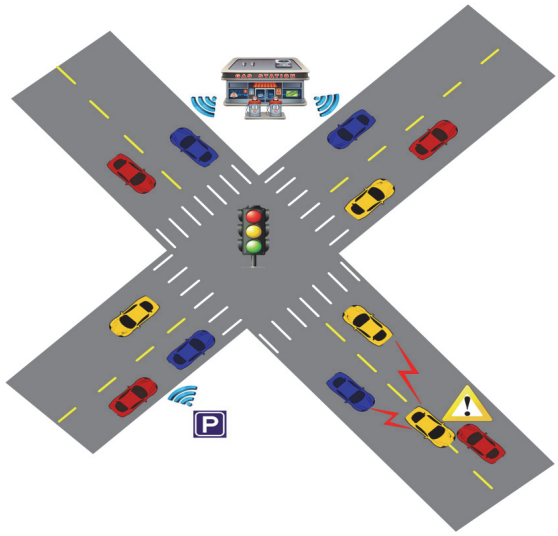


Figure 1: An instance scenario of collision warning.

how to select appropriate nodes and improve the efficiency of data propagation in vehicular networks.

In this paper, a data dissemination scheme based on fuzzy logic and network coding in vehicular networks is proposed, named SFN. A primary backbone and a slave backbone are selected in each road segment to construct a backbone network, through which all the packets are transmitted between different road segments. The backbone nodes are stable for a certain period of time and serve for different data flows. Therefore, the SFN scheme avoids too frequent relay node selection and decreases the probability of data resource contention. Specifically, the backbone nodes are selected according to the transmission abilities of all the vehicles in this segment, which are calculated by fuzzy logic and comprehensive consideration of the vehicle velocity change rate, the velocity optimization degree, and the channel quality. Additionally, based on network coding, an efficient forwarding and retransmission algorithm is designed for both intersegment and intrasegment communications. It helps in reducing and accelerating the retransmissions; therefore, the resource consumption decreases and the dissemination latency is shortened.

The main contributions of this proposal are in three aspects.

(1) Construct a backbone network composed of two backbone nodes in each road segment to support data transmissions between road segments. This backbone network utilizes fuzzy logic to select backbones based on three factors: the velocity change rate, the velocity optimization degree, and the channel quality. This is done to improve the stability and reliability of the backbone network.

(2) Network coding in data forwarding and retransmission algorithm improves the backbone-to-member and interbackbone transmissions. This improvement is because backbone nodes encode and decode the data packets, leading to a small transmission overhead and a quick data recovery.

(3) Conduct sufficient experiments to evaluate the performances of this proposal and analyze some parameters. In particular, real taxi trajectory data in Sanya, China, are used to construct vehicular scenarios. The experimental results show that SFN has a higher delivery ratio and a shorter dissemination delay than compared schemes.

The remainder of the paper is organized as follows. In Section 2, the related work on data dissemination in vehicular networks is introduced. In Section 3, an overview of SFN scheme is provided. The details of the backbone network construction are discussed in Section 4. The coding and forwarding algorithm is presented in Section 5. Then, experimental results and analysis are shown in Section 6. Finally, Section 7 provides the conclusion of this paper.

2. Related Work

Currently, a big challenge in vehicular networks is to achieve stable and reliable data transmission in the scenarios with frequent topology change and limited bandwidth. Some early researches use flooding for data transmission, in which a node rebroadcasts packets to its neighboring vehicles once it receives them. When there is a traffic jam on the road, it is easy to cause broadcast storms and information congestion, and hence the dissemination delay sharply increases. Then, some studies try to reduce the transmissions of redundant packets. In order to improve the information propagation reliability and address the broadcast storm problem, Korkmaz et al. only allow the node farthest from the sender to broadcast the packet [7]. In [8], the receiver calculates the forwarding probability based on the distance between the sender and the receiver and sets the forwarding waiting time for retransmission checking according to the current time slot. The next-hop node is selected considering a single factor, which may cause velocity instability and decrease channel quality. Shen et al. design a data scheduling framework, which avoids the collisions and improves the dissemination efficiency by providing the transmission opportunity to nodes with maximum utilities [9].

Some studies analyze the impacts of relevant factors on the dissemination performance in vehicular applications. Zhu et al. propose a data forwarding strategy based on relative velocity and distance between vehicles, which improves forwarding efficiency [10]. In [11], the vehicles parked on the pavement or in parking lots are selected as backbones, which extend the coverage of the network. A scheme in [12] introduces a delay model and an improved greedy broadcast algorithm as well as a coverage elimination rule, taking into account road topology and traffic signals. Specifically, vehicle density is considered in some routing schemes. In [13], routing protocols are adapted for vehicular applications in a real-time way, according to the current level of vehicle density. For highway or urban scenarios, a protocol supports multidirectional data dissemination by combining a generalized time slot scheme based on directional sectors and a store-carry-forward algorithm [14].

In recent years, as a classical mathematical method, fuzzy logic has been utilized to improve data propagation

in vehicular networks. In [15], a protocol incorporates fuzzy logic with geographical routing when making forwarding decisions. It takes the moving direction and the distance as the inputs of fuzzy logic and improves the delivery ratio. A seamless streaming dissemination system for vehicular networks is designed in [16]. It uses fuzzy logic to check if a roadside unit or a vehicular node can be a candidate to transfer stream data for users or not. In [17], Wu et al. propose a fuzzy-logic-based algorithm considering link quality, intervehicle distance, and vehicle mobility and design a redundancy transmission approach to enhance reliability.

Besides, since network coding can enhance data delivery in wireless communications, some existing researches focus on its benefits in data dissemination [18]. The paper [19] uses network coding to rebroadcast the messages, which improves the overall reliability and delivery ratio. In [20], cache solutions utilize network coding to reduce bandwidth cost and shorten latency. An abstract model of a general network coding process is developed to support distribution of content in vehicular networks. Furthermore, some studies combine fuzzy logic with network coding in vehicular networks. In SBN scheme, the factors including vehicle velocity, vehicle density, and channel quality are taken into account in fuzzy logic, while it also uses network coding to improve the transmission efficiency [21]. However, since this scheme prefers to select the vehicles with slow velocities as backbone nodes, its performance is greatly affected when most of the vehicles have much higher velocities than the backbones. FUZZBR scheme in [22] models the forwarding ability based on distance, velocity, and communication quality and selects two relays within a particular range by using fuzzy logic. In [23], fuzzy logic is utilized to select next-hop nodes, considering factors such as the distance between vehicles and vehicle velocity and density.

In order to improve the data dissemination in vehicular networks, how to select appropriate backbone nodes and how to fully explore network coding both require further study. Therefore, the scheme applied in this paper uses fuzzy logic to comprehensively consider the state of traveling vehicles and the channel quality of V2V communications. In addition, two backbone nodes in each road segment are selected to construct a reliable backbone network and use network coding in data forwarding to shorten propagation delay and reduce bandwidth consumption.

3. SFN Scheme Overview

In SFN, the focus is on data dissemination scenario, where the data (such as collision warning messages) have one source (such as the vehicle which has collided) and several destinations (such as the vehicles behind which might be affected). This proposed scheme aims to deliver the data from its source to all the destinations quickly with a small overhead. It is noteworthy that, with the provision of backbone network, this scheme also applies in unicast transmission applications each having one source and one destination, with some simple adjustments.

To collect real-time information of traveling vehicles, each vehicular node gets its location and velocity from onboard equipment like GPS and speedometer. It is assumed that all the vehicular nodes have the same communication radius, denoted by R. For efficient multihop data transmissions, a long road is divided into multiple segments, and the length of each segment is $R/2$, which ensures the connectivity of the backbone network [20]. It should be noted that the road width is negligible when compared with the transmission radius.

In the scheme, there are two kinds of vehicular nodes: backbone nodes and member nodes. In every segment, two backbone nodes are selected, named primary backbone and slave backbone, and other nodes are member nodes. The backbone nodes transmit packets between different road segments and propagate packets to member nodes in the same segment, while member nodes only generate and receive packets. In each segment, primary backbone has a higher transmission ability than slave backbone, and it takes more data transmission tasks. Having two backbone nodes in each segment, as opposed to one backbone node, has two main benefits. First, if one backbone loses data or gets incorrect data, the other can transmit correct data to the next segment immediately. In addition, the missing or incorrect data can be recovered through exchanging packets between backbones. In this way, the scheme reduces the transmission overhead and shortens the dissemination delay.

When a data packet such as a collision warning message is generated and is ready to be disseminated in a specific area, the sender transmits it to the backbone nodes in the same segment first. In the next step, it is transmitted through the backbone network to all the target segments. In an example scenario shown in Figure 2, a highway has four segments: S_1, S_2, S_3, and S_4. In S_1, v_2 is the primary backbone and v_3 is the slave backbone. Similarly, v_7, v_{10}, and v_{11} are primary backbones in their segments, while v_8 and v_{12} are slave backbones. The source v_1 generates two data packets p and q, which need to be broadcasted in all four segments. First, v_1 transmits the original data to the backbones v_2 and v_3 in its segment. Then, v_2 encodes them to $p + q$ and $2p + q$ and forwards the encoded packets to the backbones v_7 and v_8 in the next segment. If v_7 loses $p + q$ and v_8 loses $2p + q$, they firstly send the received packets to the next backbone node v_{10} and then recover their lost packets through packet exchange between themselves. Backbones v_{11} and v_{12} obtain the data through backbone network in a similar way. Additionally, the primary backbones v_2, v_7, v_{10}, and v_{11} decode and broadcast the data to the member nodes in their segments.

If there is only one vehicle in a segment, such as v_{10} in S_3, it is selected as a unique backbone, the primary backbone. Data dissemination in sparse scenarios will be discussed later. This paper uses linear network coding [24] in the examples and experiments; however, this suggested scheme also supports other network codes.

It is obvious that how to select backbone nodes greatly affects the data dissemination performance. In SFN, backbone nodes are selected periodically according to the transmission abilities of the vehicles. Specifically, each vehicle calculates its transmission ability and shares it with others

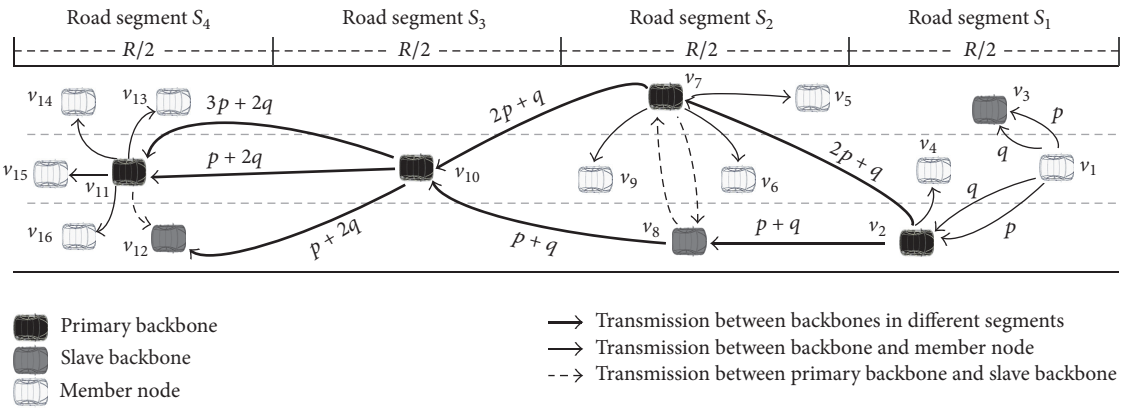


Figure 2: A scenario instance of SFN.

within the same segment. The two nodes with the highest abilities work as backbone nodes, among which primary backbone has a higher ability than slave backbone.

Here, the method to compute the transmission ability is the key for backbone selection. Since the relation between the driving status and the communication quality is not clear, it is challenging to design a simple and accurate model. Considering that fuzzy logic is an efficient method to model complicated relations [25], it is used in this study to compute the transmission ability. By doing so, three factors are taken into account: the velocity change rate, the velocity optimization degree, and the channel quality. With respect to the cycle of backbone selection, from experiment results in Section 6, the backbone network in SFN is more stable than the compared schemes. Therefore, the backbone selection cycle can be longer and hence it is less costly to maintain backbone network.

In vehicular data services, a vehicle may receive several messages that need to be forwarded to others from different neighbors. In traditional schemes, a sender sends out the original messages in different directions, regardless of the existing packets at receivers. Additionally, once several packets are lost, the sender resends these packets, not considering the different lost packets at different receivers. In this way, the network has a large retransmission overhead especially in case of lots of data. However, network coding works well in these cases, which reduces the transmission consumption. In SFN, backbone nodes encode the original packets to coded packets and hence improve the communication performance. This is explained in detail in Section 5.

In dense vehicular networks, the backbone network helps in transmitting data effectively and reducing the probability of broadcast storm. However, in sparse vehicular scenarios, it is difficult to select two backbones in every segment, and hence the data dissemination may be interrupted. In order to support data propagation in sparse areas, in SFN, if there is a unique vehicular node in a segment, it is regarded as the primary backbone to maintain connectivity. If there is no vehicle in a segment S_i, a sender in the previous segment S_{i-1} tries to find another forwarder in the next segment S_{i+1}. Since the length of each segment is $R/2$, two nodes in S_{i-1} and S_{i+1} might be in the transmission range of each other. Furthermore, if several nodes in S_{i+1} could communicate with the sender in S_{i-1}, two nodes are selected that are moving in the same direction with the sender and having the largest transmission abilities as the next backbones. Although the delivery probability may be reduced due to a long distance between nodes in S_{i-1} and S_{i+1}, the connectivity of the backbone network is enhanced. In case the vehicle density is so low that the sender cannot communicate with any vehicles in the next two segments, it carries packets and continues traveling until a next-hop vehicle appears.

4. Backbone Network Construction

In this section, the selection of backbone nodes and construction of a backbone network are discussed. The backbone node selection is decided by the transmission abilities of the vehicular nodes. These abilities are computed according to three transmission factors: the velocity change rate, the velocity optimization degree, and the channel quality. These factors are not independent; for example, a large velocity change rate leads to a low channel quality. Considering the complicated relations among these factors and the transmission ability, fuzzy logic is utilized to model the transmission ability. In detail, there are four steps as listed below.

(1) Calculation of the Transmission Factors. Each vehicle gets its driving and communication information from its sensors and hello messages and then calculates its velocity change rate, velocity optimization degree, and channel quality.

(2) Fuzzification. For each factor, a membership function is used to convert an original value to several fuzzy values.

(3) Calculation of the Transmission Ability Rank. All the combinations of nonzero fuzzy values of the three factors are obtained. For each combination, it is mapped to transmission ability ranks according to preset rules.

(4) Defuzzification. A function graph and a defuzzification method are designed to convert the transmission ability ranks and values to an overall transmission ability value.

4.1. Transmission Factors. In vehicular networks, the vehicle velocity and the channel quality greatly affect the transmission performance. Here, three variables are introduced to represent relevant factors.

(1) Velocity Change Rate. The relative velocity is an important factor in vehicular networks [26]. High velocity jitter usually leads to intermittent communication and transmission failures accordingly. Consequently, the vehicle with a high velocity jitter is not suitable to be selected as a backbone node. In order to indicate the velocity jitter of a vehicle, the velocity change rate is introduced, denoted by vc and calculated by

$$vc = \left| \frac{v \cdot vl}{vs^2} - 1 \right|. \quad (1)$$

Here, v is the current velocity of the vehicle, vs is the average velocity within a recent short duration of t_s, and vl is the average velocity in a recent long period of time t_l ($t_l > t_s$).

Since vs and vl imply the recent mobility features of the vehicle and v shows the current state, vc indicates the recent velocity change rate. Because of the frequent velocity changes, here two average velocities are used rather than velocities at two specific times in the past. In (1), a big difference between vs and vl or between v and vs, meaning a high velocity jitter, results in a large vc. It is apparent that the larger the value of vc is, the less chance the vehicle has to be selected as backbone. Therefore, in order to easily use fuzzy logic, only those vehicles with vc in [0, 1] are considered as backbone candidates. This study's experiments show that SFN performs well when $t_l = 10 \times t_{\text{hello}}$ and $t_s = 4 \times t_{\text{hello}}$, where t_{hello} is the hello message cycle.

(2) Velocity Optimization Degree. The velocity change rate factor is not enough for backbone selection. This is because a vehicle with a stable velocity may not be a good backbone if its speed is very different from other vehicles in the same segment. The velocity optimization degree shows the relation between a vehicle's average velocity and the optimal velocity of vehicles in the same segment, denoted by vd and calculated by

$$vd = \left| 1 - \left| \frac{vl}{V} - 1 \right| \right|, \quad vl \le 2V. \quad (2)$$

Here, V is the optimal velocity of vehicles in this segment at this time.

Only the vehicles whose velocity vl is less than or equal to $2V$ are considered to be backbone candidates. In this way, the value domain of vd is [0, 1], which can be easily analyzed in fuzzy logic, and a vehicle with a similar velocity to V has a high vd. In other words, a high vd means the vehicle should have a high probability to be backbone. This is because it keeps the same pace with other vehicles and has a long encountering time to transmit data. Meanwhile, the above equation can be represented by

$$vd = \begin{cases} \dfrac{vl}{V}, & vl \le V; \\ 2 - \dfrac{vl}{V}, & V < vl \le 2V. \end{cases} \quad (3)$$

There are many factors affecting the vehicle velocity, and [27] proposes an optimal velocity model by investigating the properties of congestion and the delay time of car motion. The optimal velocity V is determined by the vehicle density, the number of lanes, the traffic accidents, and some other factors. Although it is a complicated issue to select an appropriate value for V, the comprehensive analysis of present and predicted traffic information may give some clues. For further information, please refer to [28, 29]. In this study's simulation, the same optimal velocity is set for all segments except the two with smaller values, due to assumed traffic collisions.

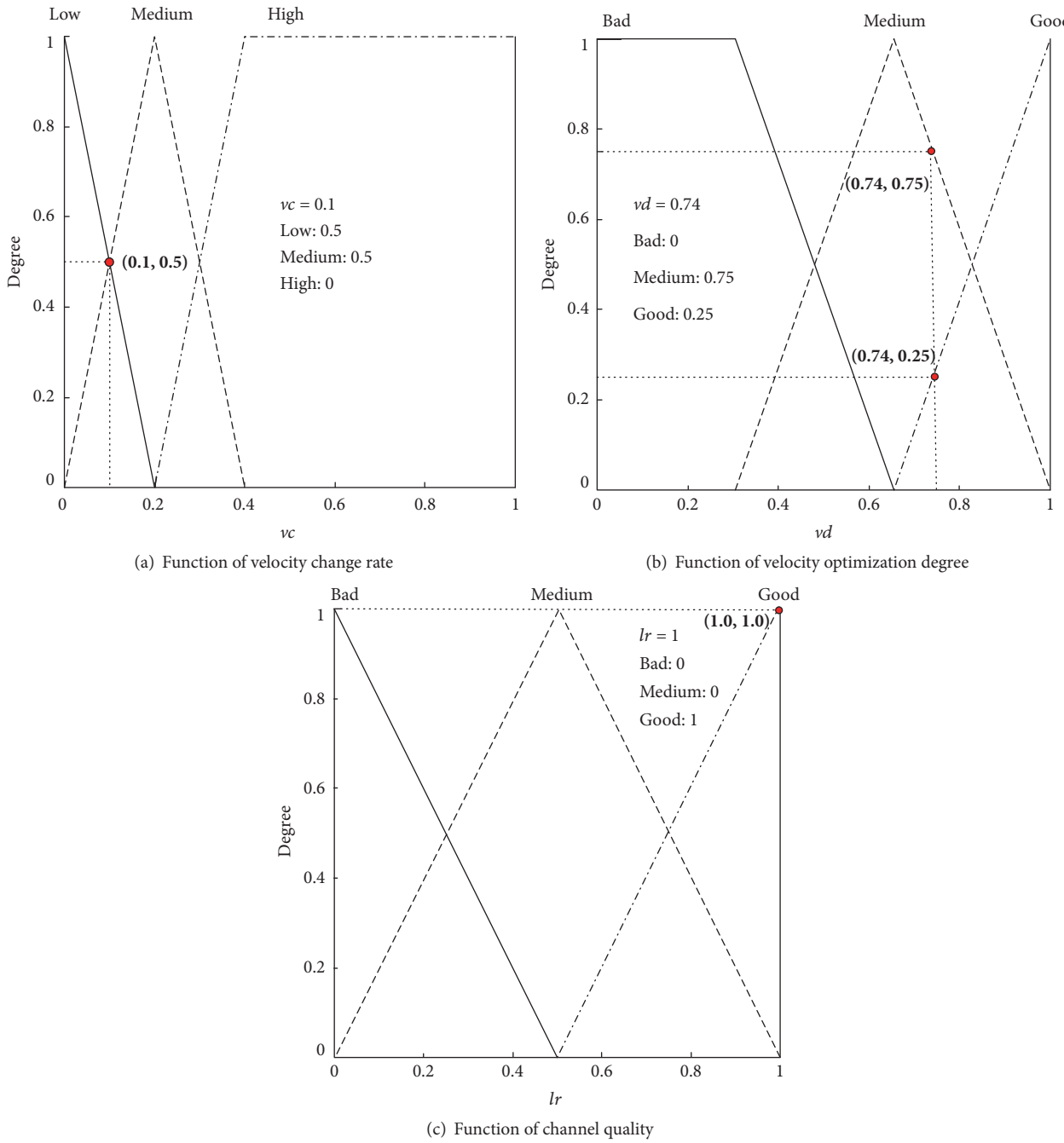
(3) Channel Quality. The channel quality reflects the reliability of intervehicle channel, denoted by lr. It is related to many factors, including the network technology, the local environment, the signals traveling through channel, and the fundamental physics behind wireless transmission. A greater channel quality provides a higher delivery probability. It is difficult to estimate channel conditions in vehicular networks accurately due to the frequent changes of network topology and the complex environmental factors, such as weather and nearby buildings [30]. Thus, the delivery ratio of hello messages, which are periodically exchanged among all the vehicles in the same segment, is used in experiments to represent channel quality.

4.2. Fuzzification. A membership function presents whether a value of an element falls within a specific range and indicates the membership degree in a fuzzy set [31]. The fuzzy set is obtained by assigning a value to each level to represent its grade of membership function. In SFN, fuzzy logic uses a membership function to convert the value of every factor to a fuzzy set. Suitable membership functions are acquired for transmission factors through data analysis and simulation experiments, shown in Figure 3. For instance, if the usual velocity range is [60, 100] km/h, an extreme value of the velocity optimization degree is got when $vl = 100$ and $V = 60$ as $vd \approx 0.3$. Therefore, in the membership function of vd, (0, 0.3) is "bad" with probability of 1. Some additional experiments with different membership functions are illustrated in Section 6.3.

As Figure 3 shows, the velocity change rate has three levels {Low, Medium, High}, while the velocity optimization degree and the channel quality both have three levels {Bad, Medium, Good}. In Figure 3(a), when the velocity change rate vc is 0.1, its fuzzy set via mapping is {Low: 0.5, Medium: 0.5, High: 0}. Similarly, when the velocity optimization degree vd is 0.74 and the channel quality lr is 1, their fuzzy sets are {Bad: 0, Medium: 0.75, Good: 0.25} and {Bad: 0, Medium: 0, Good: 1}, respectively.

Based on the three fuzzy sets, several combinations are obtained, each of which consists of three nonzero fuzzy values relevant with three factors. In the above example shown in Figure 3, four combinations are listed in Table 1.

4.3. Transmission Ability Rank. First, the influences of transmission factors on the transmission ability in theory as well



(a) Function of velocity change rate

(b) Function of velocity optimization degree

(c) Function of channel quality

Figure 3: Membership functions of transmission factors.

Table 1: Fuzzy value combinations.

Combination	*vc*	*vd*	*lr*
c_1	Low: 0.5	Medium: 0.75	Good: 1
c_2	Low: 0.5	Good: 0.25	Good: 1
c_3	Medium: 0.5	Medium: 0.75	Good: 1
c_4	Medium: 0.5	Good: 0.25	Good: 1

as the feedback from experiments are analyzed. In the next step, 27 rules are set to determine the transmission ability rank, denoted by *tar*, as listed in Table 2. According to the rules, map the fuzzy sets of three transmission factors into a transmission ability rank. Overall, there are six ranks of the transmission ability as {Perfect, Good, Acceptable, Not Acceptable, Bad, Very Bad}.

Each fuzzy value combination is mapped to a transmission ability rank according to the rules. Meanwhile, the rank value is the minimum fuzzy value in the combination. When multiple combinations are mapped to the same rank with different rank values, the maximum rank value is selected. To sum up, max-min method [32] is used to calculate

Table 2: The rules of transmission ability ranks.

Rule	*vc*	*vd*	*lr*	*tar*
Rule 1	Low	Good	Good	Perfect
Rule 2	Low	Good	Medium	Good
Rule 3	Low	Good	Bad	Not Acceptable
Rule 4	Low	Medium	Good	Good
Rule 5	Low	Medium	Medium	Acceptable
Rule 6	Low	Medium	Bad	Bad
Rule 7	Low	Bad	Good	Not Acceptable
Rule 8	Low	Bad	Medium	Bad
Rule 9	Low	Bad	Bad	Very Bad
Rule 10	Medium	Good	Good	Good
Rule 11	Medium	Good	Medium	Acceptable
Rule 12	Medium	Good	Bad	Bad
Rule 13	Medium	Medium	Good	Acceptable
Rule 14	Medium	Medium	Medium	Not Acceptable
Rule 15	Medium	Medium	Bad	Bad
Rule 16	Medium	Bad	Good	Bad
Rule 17	Medium	Bad	Medium	Bad
Rule 18	Medium	Bad	Bad	Very Bad
Rule 19	High	Good	Good	Not Acceptable
Rule 20	High	Good	Medium	Bad
Rule 21	High	Good	Bad	Very Bad
Rule 22	High	Medium	Good	Bad
Rule 23	High	Medium	Medium	Bad
Rule 24	High	Medium	Bad	Very Bad
Rule 25	High	Bad	Good	Bad
Rule 26	High	Bad	Medium	Very Bad
Rule 27	High	Bad	Bad	Very Bad

Table 3: Combination mapping results.

Combination	Rule	*tar*	Rank value
c_1	Rule 4	Good	0.5 [= min (0.5, 0.75, 1)]
c_2	Rule 1	Perfect	0.25 [= min (0.5, 0.25, 1)]
c_3	Rule 13	Acceptable	0.5 [= min (0.5, 0.75, 1)]
c_4	Rule 10	Good	0.25 [= min (0.5, 0.25, 1)]

Table 4: Final transmission ability ranks.

tar	Value
Perfect	0.25
Good	0.5 [= max(0.5, 0.25)]
Acceptable	0.5

the transmission ability rank. For the above example, the combination mapping results are shown in Table 3, and the final transmission ability ranks are listed in Table 4.

4.4. Defuzzification. Next, several transmission ability ranks are converted into a digital number, which is the transmission ability value, denoted by *tav*, according to a function graph. The function graph of SFN scheme is shown in Figure 4. *tav* is the coordinate of the center of gravity of a shadow *sp*, which

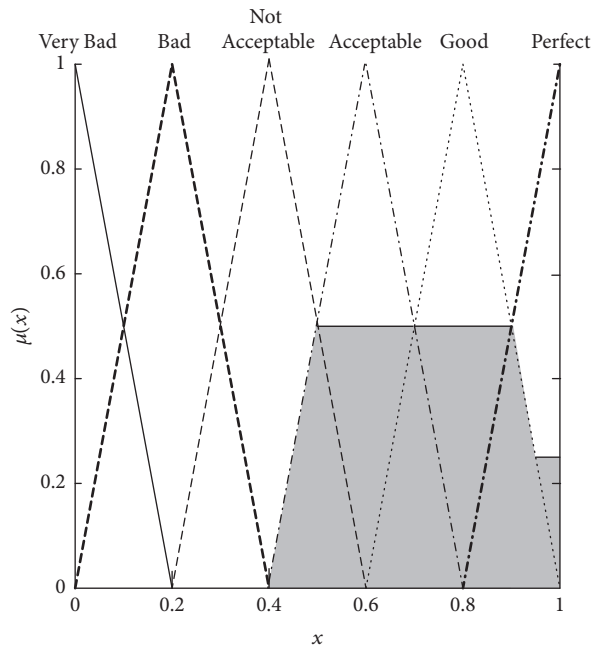


Figure 4: Function graph of transmission ability rank.

is determined by the ranks and their values. *tav* is calculated by

$$tav = \frac{\int_{x \in sp} \mu(x)\, x\, dx}{\int_{x \in sp} \mu(x)\, dx}. \tag{4}$$

Still in the above example, the shadow in Figure 4 is in line with the transmission ability ranks listed in Table 4. Accordingly, the transmission ability value $tav = 0.71$ is calculated.

Each node broadcasts its own transmission ability value to others in its road segment. The higher the transmission ability value is, the better the node works as a backbone. Therefore, every node sorts the received values in ascending order and selects the nodes with maximum values as backbones. For instance, in Figure 2, there are six vehicles in the segment S_4, which are v_{11}, v_{12}, v_{13}, v_{14}, v_{15}, and v_{16}, and their transmission ability values are 0.91, 0.88, 0.75, 0.85, 0.66, and 0.76, respectively. Each vehicle receives and sorts the transmission ability values and selects v_{11} as primary backbone and v_{12} as slave backbone.

5. Data Coding and Forwarding Algorithm

In data transmission process, SFN uses network coding to reduce bandwidth consumption and support fast recovery when packet loss occurs. There are two main cases as follows.

(1) In wireless communications, when there are several packets to be forwarded in different directions, the relay node can send a small number of coded packets to complete the delivery. As shown in Figure 5, v_1 and v_3, respectively, send packets q and p to each other through a relay node v_2. Without network coding, v_2 needs to send q to v_3 and p to v_1,

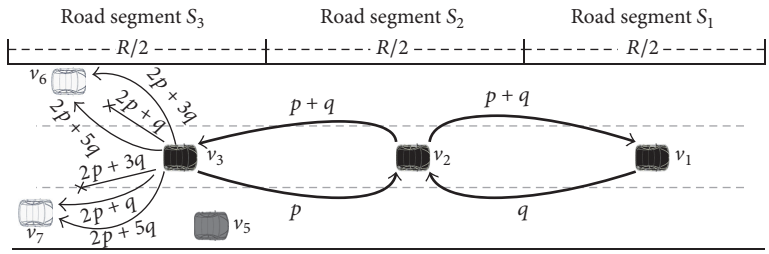


Figure 5: An example of data coding and forwarding.

and thus the total number of transmissions is 4. By contrast, with network coding, v_2 only needs to broadcast $p + q$ to v_1 and v_3. Hence, the total number of transmissions is 3. This example shows that network coding reduces the transmission overhead which results in improvement of the bandwidth utilization.

(2) Network coding also works for efficient data recovery. In Figure 5, a primary backbone v_3 wants to broadcast two original packets p and q to its member nodes. First, v_3 encodes p and q to coded packets $2p + q$ and $2p + 3q$, which are linearly independent. If v_6 only receives $2p + 3q$ and v_7 only receives $2p + q$ correctly, v_3 needs to resend data to v_6 and v_7. Using networking coding, v_3 does not need to resend $2p + q$ and $2p + 3q$ again. Instead, it creates and sends a new coded packet $2p + 5q$ with a new coefficient matrix. Then, v_6 and v_7 individually decode two coded packets and get original packets. This example demonstrates the improvement in retransmission efficiency and saving network resources through network coding.

In SFN, every primary backbone uses network coding to transmit data packets to backbones in the next segment and member nodes in the same segment, while each slave backbone only transmits coded packets to backbone nodes. After receiving data packets, the primary backbone decodes them to recover the original data. Then, it encodes these packets with its own coefficient matrix and finally broadcasts them to member nodes in the same segment and backbones in the next segment. If some packets are lost, the receiver sends a request message to the backbones in the same segment or in the upstream segment. In the next stage, when another node receives this request, it encodes its packets and transmits them to the request sender. An example of data coding and forwarding of backbone B_i^P in a segment S_i, when it receives a packet p, is illustrated in Algorithm 1.

In the algorithm, p is the packet B_i^P just received, and OP_i^P and CP_i^P are the original packets and the coded packets carried by B_i^P, respectively. NOP is the new original packets B_i^P gets by decoding p and other existing packets; DP is the new coded packets B_i^P creates; B_i is the backbones in the segment S_i, consisting of primary backbone B_i^P and slave backbone B_i^S; and M_i is the member nodes in the same segment.

Also, $\mathrm{SOP}(x)$ is the original packets used to generate a coded packet x; $\mathrm{YP}(x)$ is the existing packets at B_i^P, which are generated by all or part of original packets in x; and $\mathrm{DIR}(x)$ is the forwarding direction of packet x. There are two forwarding directions along the road; $\mathrm{DIR}(x) = 1$ means data transfer from S_i to S_{i+1} and $\mathrm{DIR}(x) = -1$ means data transfer from S_i to S_{i-1}. In addition, $\mathrm{REQ}(x)$ is a request for data packet x.

According to the data coding and forwarding algorithm, when B_i^P receives a data packet p correctly, it tries to get original packets NOP. Then, B_i^P generates and forwards coded packets of NOP to the next-hop backbones and delivers them to the member nodes. If the data packet is not received correctly, B_i^P sends a request of this data to its slave backbone B_i^S for data sharing. When B_i^P receives a request of some data, if it has that data, it encodes and replies to the request sender; otherwise, it forwards the request to the last-hop backbones.

In SFN, fuzzy logic is used to select two backbone nodes in each road segment, construct a backbone network, and use network coding to propagate data packets to target road segments. The backbone nodes with suitable velocity and good channel quality help to improve the transmission efficiency, while the network coding in intrasegment broadcast and intersegment forwarding may enhance the utilization of limited communication resources.

6. Performance Evaluation

6.1. Network Configurations. In order to evaluate the performance of SFN scheme, the opportunistic network environment simulator (ONE [33]) is used to conduct simulation experiments. The scenario configurations are listed in Table 5. Regarding the mobility model, a vehicle-following model [34] is utilized, which results in different vehicle velocities and different densities in the road segments. For *vd* calculation, to simulate different optimal velocities in different segments, a small optimal velocity of 60 km/h for two segments is set, 300–450 m and 1350–1500 m, while other segments have a higher optimal velocity of 90 km/h. Moreover, the influences of different backbone selection cycles, velocity ranges, and membership functions of transmission factors are discussed in Section 6.3.

SBN [21] and FUZZBR [22] are chosen as compared data dissemination schemes in vehicular networks. Compared with SBN, the proposed scheme SFN has several advantages. Although both of them utilize backbone nodes to forward data, SFN improves the backbone selection and the backbone network construction. Firstly, SBN prefers a vehicle with slow speed to be backbone in a road segment, no matter

Input: p, OP_i^P, CP_i^P;
Output: NOP, DP;
(1) **if** p is a correct data packet **then**
(2) $uop = \mathrm{SOP}(p) - \mathrm{OP}_i^P$;
(3) **if** $\exists sc \subseteq \mathrm{YP}(uop)$ satisfying decoding conditions **then**
(4) obtain original packets $\mathrm{SOP}(sc)$ by decoding sc;
(5) $\mathrm{NOP} = \mathrm{SOP}(sc) - \mathrm{OP}_i^P$;
(6) **while** $\exists\{r, s\} \subseteq \mathrm{NOP}$ having $\mathrm{DIR}(r) \neq \mathrm{DIR}(s)$ **do**
(7) create a coded packet $\mathrm{DP}(\{r, s\})$;
(8) send $\mathrm{DP}(\{r, s\})$ to B_{i+1} and B_{i-1};
(9) send two coded packets using $\{r, s\}$ to M_i;
(10) $\mathrm{NOP} = \mathrm{NOP} - \{r, s\}$;
(11) **end**
(12) create coded packets DP(NOP);
(13) **if** $\mathrm{DIR}(\mathrm{NOP}) = 1$ **then**
(14) send DP(NOP) to B_{i+1} and M_i;
(15) **else**
(16) send DP(NOP) to B_{i-1} and M_i;
(17) **end**
(18) **end**
(19) **end**
(20) **if** p is an incorrect data packet **then**
(21) send $\mathrm{REQ}(p)$ to B_i^S;
(22) **end**
(23) **if** p is $\mathrm{REQ}(q)$ from node z **then**
(24) **if** q **is carried by** B_i^P **then**
(25) **if** $\exists$ other original packet t to be sent to z **then**
(26) create one or several coded packets $\mathrm{DP}(\{q, t\})$;
(27) send $\mathrm{DP}(\{q, t\})$ to z;
(28) **else**
(29) send q to z;
(30) **end**
(31) **else**
(32) **if** $\mathrm{DIR}(q) = 1$ **then**
(33) send $\mathrm{REQ}(q)$ **to** B_{i-1};
(34) **else**
(35) send $\mathrm{REQ}(q)$ **to** B_{i+1};
(36) **end**
(37) **end**
(38) **end**

Algorithm 1: Data coding and forwarding algorithm for primary backbone B_i^P.

Table 5: Simulation environment configurations.

Parameter	Value
Road	2000 m with 4 lanes
Number of vehicular nodes	20, 40, 60, 80, 100
Data packet size	512 bytes
Vehicle velocity	Random in [60, 100] km/h
Communication radius	300 m
Backbone selection cycle	4 s
Simulation time	150 s
Hello packet exchange cycle	1 s

how fast other vehicles drive; SFN chooses a vehicle with a speed close to the optimal speed in each segment. Therefore, the backbones in SFN are probably in a similar pace to others, which results in more intervehicle communication chances. Secondly, considering that the frequent change of velocities affects the quality of wireless transmission, SFN chooses vehicles with relatively stable speeds as backbones, which enhance the efficiency of data delivery. Thirdly, SFN uses fuzzy logic to calculate the transmission abilities of vehicles, as well as a large quantity of experiments to figure out the appropriate membership functions. Fourthly, for sparse vehicular scenarios, SFN maintains the connectivity of backbone network by picking up backbones from a farther segment rather than the next one. Last but not least, while SBN has only one backbone in each segment, there are two backbones, primary backbone and slave backbone, in each segment in SFN, which not only improve the reliability of backbone network but also accelerate data delivery as a result

of network coding. Overall, SFN is an innovative scheme with several advantages over SBN.

Comparing SFN and FUZZBR, both of them utilize fuzzy logic to select nodes for forwarding. However, the factors and their membership functions in fuzzy logic are different, and the tasks of those selected nodes also differ. On the one hand, as it was discussed above, SFN uses the speed factor in depth, which helps to select appropriate backbone nodes. On the other hand, FUZZBR selects the relays for each data delivery requirement, while SFN constructs a backbone network for all the data services during a period of time, which reduces the overhead of forwarder selection and improves the efficiency of data dissemination.

The experiments evaluate four criteria: the delivery ratio, the number of transmissions, the dissemination delay, and the backbone stability. The delivery ratio is the ratio of the number of delivered packets to the total number of transmissions during the data dissemination from the source to the destination segments. The higher the delivery ratio is, the better the performance the scheme has. Besides, the number of transmissions counts the successful and failed transmissions as well as the retransmissions, which indicates the communication overhead of the scheme. Additionally, the dissemination delay is the delay from the time the data generated to the time data dissemination finishes. A scheme with a short delay works well in delay-sensitive applications. The backbone stability is the ratio of the number of nodes working as backbones during a specific period of time to the total number of backbones. Higher backbone stability implies a less frequent backbone reselection and hence a smaller update cost of backbone network.

6.2. Simulation Results. This study's SFN scheme is compared with SBN and FUZZBR, and the results are shown in Figure 6.

(1) Delivery Ratio. In Figure 6(a), the data delivery ratios of three schemes rise when the vehicle density increases, and SFN keeps a high and relatively stable ratio compared with SBN and FUZZBR. In a sparse network with 20 vehicles, SFN has a higher ratio than SBN and FUZZBR by about 25%, due to the data forwarding between two segments S_i and S_{i+2} when no vehicles exist in S_{i+1}. In the scenario with a high density (100 vehicles), the delivery ratio of SFN is also higher than others by approximately 10%, because of its stable and efficient backbone network.

(2) Number of Transmissions. As Figure 6(b) shows, when the node density is low, SFN and FUZZBR transmit less data than SBN, and vice versa. As discussed above, SFN works well when there are few nodes, while in other schemes the sender searches for the next-hop relay all the time, resulting in a large transmission overhead. In Figure 6(b), SBN has the smallest transmission overhead among the compared schemes, when there are more than 40 vehicles. The main reasons lie in two aspects. First, because of the relatively even vehicle distributions in all segments, the intervehicle communication is relatively reliable. A small number of retransmissions make the advantages of two backbones and network coding in SFN not apparent. Second, in dense scenarios, the backbone network in SBN, consisting of one backbone in each segment, is well connected. Since it avoids the exchange among backbones in the same segment, the number of transmissions is smaller than SFN. However, if the number of vehicles is a bit less, the transmission overhead of SBN rises sharply. However, SFN keeps a relatively small communication cost, despite data sharing between the primary and slave backbones, which may be acceptable in most vehicular applications.

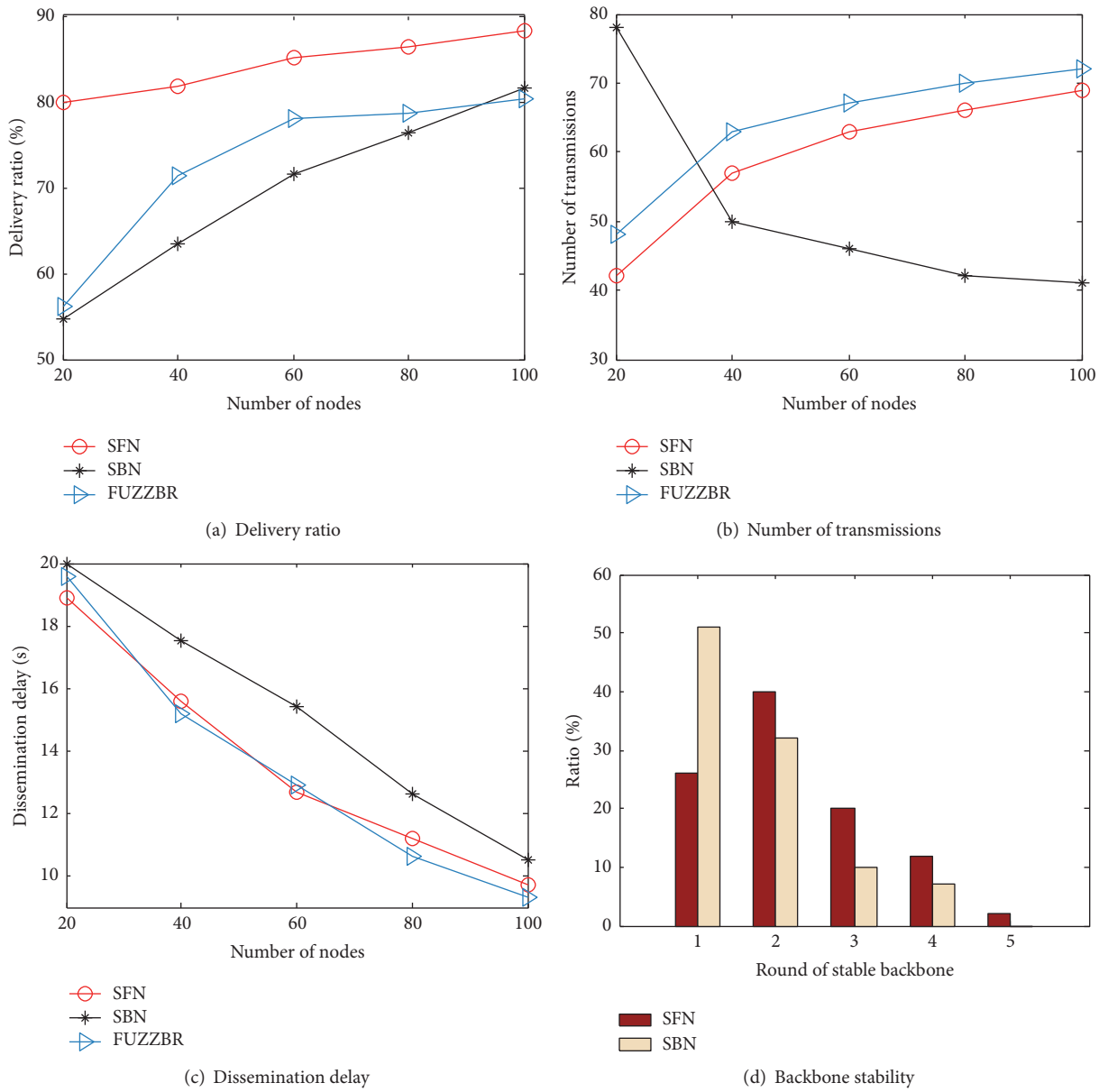
(3) Dissemination Delay. According to Figure 6(c), in sparse networks, the propagation latencies of all the schemes are longer than those in dense networks, due to the fewer encounter chances between vehicles. SBN has the longest delay, because the unique backbone in every segment has to wait for retransmissions in case of packet loss, before forwarding to the next segment. In general, SFN has a short delay, similar to FUZZBR.

(4) Backbone Stability. SFN and SBN use backbone nodes to forward the data packets, which help to avoid broadcast storms. Therefore, the maintenance of backbone network affects the performances of these schemes. The results of backbone stability in SFN and SBN are illustrated in Figure 6(d). This figure shows the proportions of nodes selected as backbones in consecutive 1–5 rounds (a round is 1 s) in all the backbones. It is obvious that, in SBN scheme, half of the backbone nodes only work for one round, which indicates that a majority of the backbone nodes are changed frequently. In comparison, the backbones in SFN are more stable than SBN. In detail, over 70% of backbones work for 2 to 4 rounds. Therefore, the cost of backbone update in SFN is small.

To sum up, in vehicular scenarios with different vehicle densities, SFN keeps a higher delivery ratio and a shorter propagation delay than SBN and FUZZBR and maintains an acceptable transmission overhead and a good stability of the backbone network.

6.3. Parameter Analysis. Considering that the backbone selection cycle and the velocity distribution affect the performances of data dissemination schemes, in this section, two main criteria are analyzed: the delivery ratio and the dissemination delay. Additionally, in order to select appropriate membership functions of transmission factors (as shown in Figure 3), a large number of experiments are conducted. Due to space limit, here only the coordinates of the peaks in medium levels of the membership functions are presented. It should be mentioned that, in these experiments, the number of vehicles is always 80.

In the first group of experiments, the backbone selection cycle ranges from 1 s to 6 s, and the results are illustrated in Figure 7. As shown in Figure 7(a), when the backbone selection cycle is shorter than 4 s, the delivery ratio of SFN keeps stable at around 87%. However, longer cycles such as 5 s and 6 s lead to lower delivery ratios by nearly 4% and 8%, because the network topology changes a lot during a long time and the backbones are not always suitable. In SBN, when the backbone selection cycle increases, the delivery ratio decreases gradually. In Figure 7(b), the dissemination



(a) Delivery ratio

(b) Number of transmissions

(c) Dissemination delay

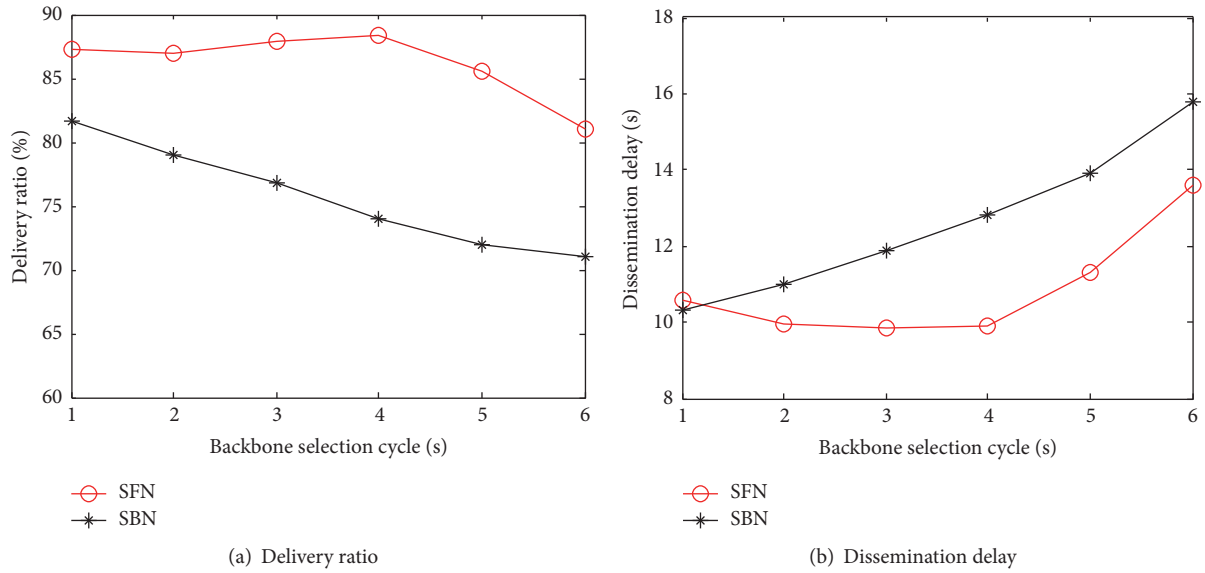
(d) Backbone stability

Figure 6: Experimental results.

delay of SFN drops slightly when the cycle changes from 1 s to 4 s and then rises with increasing cycle. The reason is that too frequent backbone selection leads to a large control cost, and too long cycle has detrimental impacts on the performance of backbone network as a result of unsuitable backbones. Overall, in order to achieve a high delivery ratio and a short dissemination delay, 4 s is a good choice for backbone selection cycle in SFN.

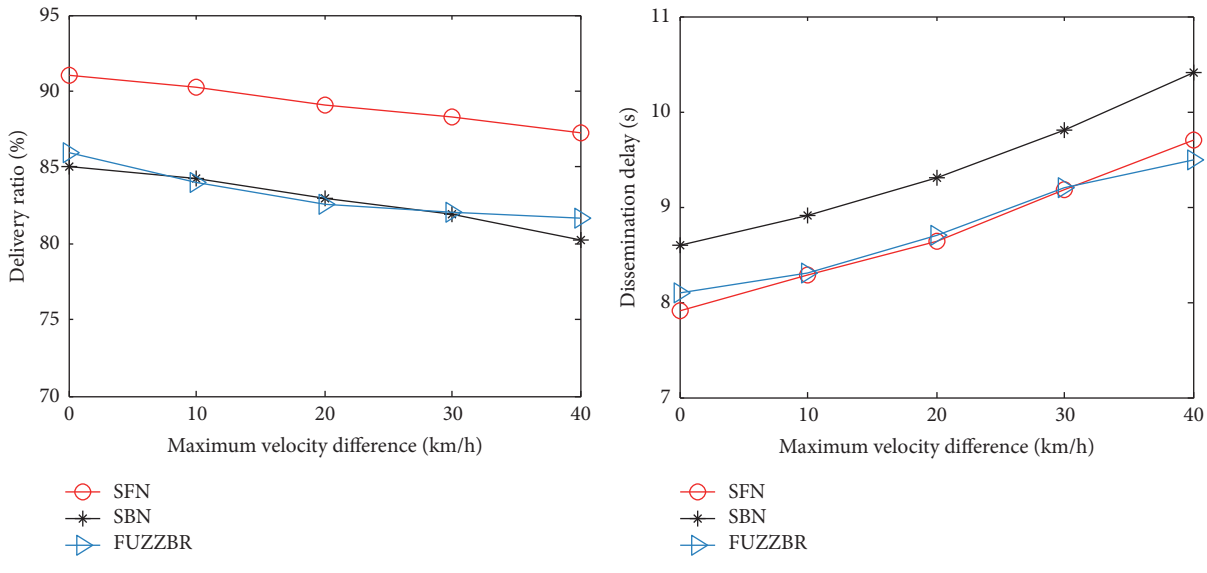
In the second series of experiments, the maximum velocity difference ranges from 0 to 40 km/h with 80 km/h as the standard velocity. In other words, the speed ranges are [80, 80], [75, 85], [70, 90], [65, 95], and [60, 100], respectively. The results are shown in Figure 8. Obviously, with a large speed range, the delivery ratio reduces and the dissemination delay grows, because the variety of mobility aggravates the unreliable wireless channels. It is noteworthy that when the speed range changes, SFN keeps a higher delivery ratio than SBN and FUZZBR, while keeping a short delay.

In the third group of experiments, the performances of SFN with different membership functions of the three factors *vc*, *vd*, and *lr* are listed in Table 6. In Figure 3, the values of *vc*, *vd*, and *lr* at the peak points in medium levels are 0.2, 0.65, and 0.5, respectively. Here, the results with other values are illustrated. Table 6 shows that, in general, the delivery ratio, the transmission overhead, and the dissemination delay are best when using this paper's selected values. Actually, a series of experiments with typical values provide guidance to determine the assignment of complicated variables.



(a) Delivery ratio (b) Dissemination delay

Figure 7: Results with different backbone selection cycles.



(a) Delivery ratio (b) Dissemination delay

Figure 8: Results with different maximum velocity differences.

Table 6: Results of SFN with different membership functions of transmission factors.

Parameters in $\{vc, vd, lr\}$		Delivery ratio (%)			Number of transmissions			Dissemination delay (s)		
$\{x, 0.65, 0.5\}$	x	0.2	0.3	0.4	0.2	0.3	0.4	0.2	0.3	0.4
	Results	86.53	83.31	73.26	59.20	72.40	65.40	9.86	10.36	11.53
$\{0.2, y, 0.5\}$	y	0.6	0.65	0.7	0.6	0.65	0.7	0.6	0.65	0.7
	Results	80.25	87.50	79.67	64.00	58.20	59.00	11.37	9.86	13.51
$\{0.2, 0.65, z\}$	z	0.4	0.5	0.6	0.4	0.5	0.6	0.4	0.5	0.6
	Results	83.60	86.26	78.55	68.20	61.00	66.40	12.1	10.8	9.97

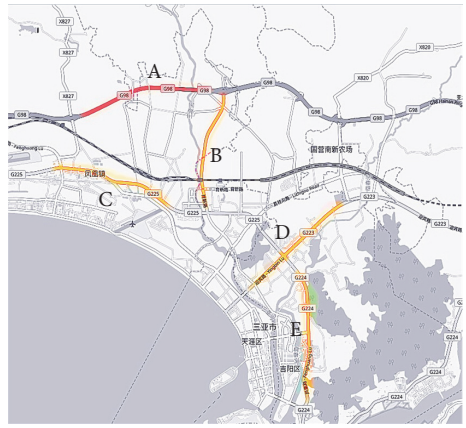


Figure 9: Selected roads in map of Sanya, China.

In one word, an appropriate backbone update frequency and suitable membership functions affect the overall performance of SFN, which can be selected through sample analysis in preliminary experiments. Moreover, SFN performs better than compared schemes in the scenarios with different speed ranges.

6.4. Performance Evaluation Using Real Taxi Trajectories. Besides the simulated vehicular scenarios, a set of experiments are carried out based on real taxi trajectories. The open-access dataset is provided by the Ministry of Transport of China [35]. It includes the real-time trajectories of 4600 taxies in Sanya, Hainan Province, from 9 a.m. to 10 a.m. on November 15, 2016. The GPS data collection cycle is 10 s.

In the experiments, to improve the data reliability, five roads are selected to create data dissemination scenarios, denoted by *A*, *B*, *C*, *D*, and *E*. They are displayed in Figure 9, and the time period is from 9 a.m. to 9:10 a.m. In these scenarios, there are 15, 14, 20, 255, and 15 vehicles traveling on the selected roads *A*~*E*, respectively. For some missing data, estimated locations according to the existing prior and next entries as well as the velocities are inserted.

The main difference between the simulation scenarios and the real trajectory scenarios is that the vehicle distributions in simulations are relatively even in all segments, while the vehicles are unevenly distributed in real scenarios. Therefore, the performances of the three schemes in real scenarios vary from the simulation results.

The results are shown in Figure 10. Compared with the results in simulated scenarios, generally, SFN can be seen to still perform best, and SBN has a better performance than FUZZBR. This is because the nonuniform traffic distribution and the different mobility patterns of vehicles greatly degrade the performance of FUZZBR.

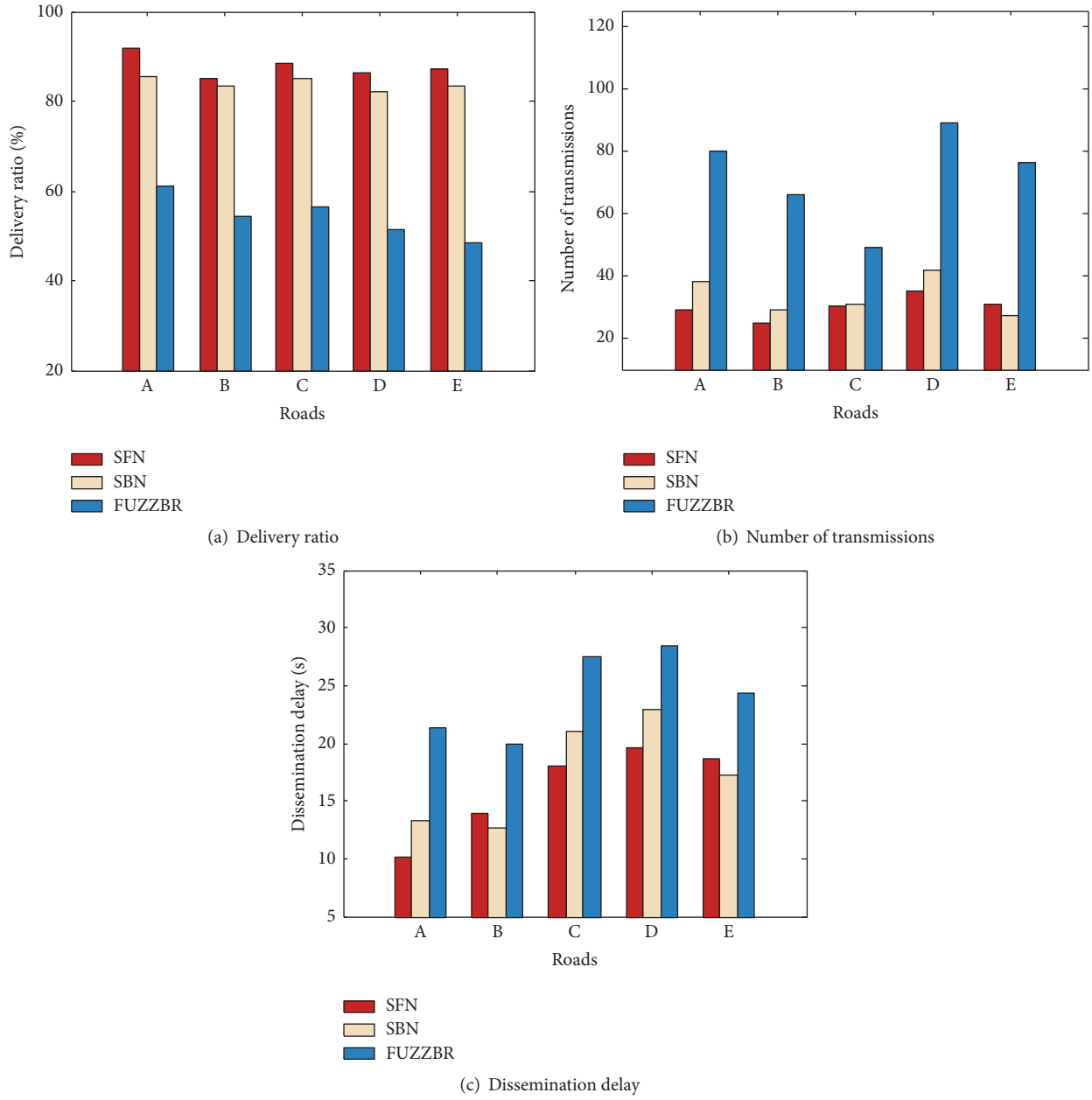
From Figure 10(b), compared with the simulation results in Figure 6(b), SBN does not show the smallest transmission overhead in most of the cases in real trajectory experiments. The reason is that a large variety of mobility patterns of vehicles leads to unstable communications and hence weakens the performance of SBN. Meanwhile, because of the reliable backbone network and the network coding, SFN shows obvious advantages over others.

In conclusion, SFN with a reliable backbone network composed of two backbones in each segment and network-coding-based data forwarding has a high delivery ratio and a short dissemination delay, in both simulated vehicular networks and real vehicular scenarios based on taxi trajectory data.

7. Conclusion

In order to improve data dissemination in vehicular networks, the proposed SFN scheme utilizes fuzzy logic to construct backbone network and network coding for efficient data forwarding. In each road segment, a primary backbone and a slave backbone are selected according to the transmission abilities of vehicles, which are calculated by fuzzy logic and take into account three transmission factors: the velocity change rate, the velocity optimization degree, and the channel quality. Then, these backbones construct a backbone network to support intersegment data dissemination. Moreover, when transmitting several packets in different directions or retransmitting more than one packet, SFN uses network coding to reduce the transmission overhead and hence saves wireless bandwidth and achieves quick recovery. In particular, for sparse scenarios, a specific solution is put forward to accelerate data propagation. The experimental results show that SFN has a higher delivery ratio and a shorter dissemination delay than other schemes, while keeping backbone stability.

However, the quality enhancement of wireless communications in light of MAC and physical layers could further improve the performance of data dissemination schemes [36]. Furthermore, to model the transmission ability of vehicles thoroughly in theory will be a challenging but significant attempt in the future.



(a) Delivery ratio

(b) Number of transmissions

(c) Dissemination delay

Figure 10: Results using real taxi trajectories.

Conflicts of Interest

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

Acknowledgments

The authors gratefully acknowledge the support from the National Natural Science Foundation of China (61502320 and 61373161), Science & Technology Project of Beijing Municipal Commission of Education in China (KM201410028015), Youth Backbone Project of Beijing Outstanding Talent Training Project (2014000020124G133), and Cultivation Object of Young Yanjing Scholar of Capital Normal University.

References

[1] M. Hu, Z. Zhong, M. Ni, and A. Baiocchi, "Design and analysis of a beacon-less routing protocol for large volume content dissemination in vehicular ad hoc networks," *Sensors (Switzerland)*, vol. 16, no. 11, article 1834, 2016.

[2] X. Tang, J. Pu, K. Ma, and Z. Xiong, "Cooperative transmission control scheme using erasure coding for vehicular delay-tolerant networks," *Journal of Supercomputing*, vol. 68, no. 3, pp. 1462–1486, 2014.

[3] X. L. Tang, D. H. Hong, W. L. Chen, and J. H. Pu, "Distributed storage scheme using bipartite graph matching for vehicular networks," *Journal of Software. Ruanjian Xuebao*, vol. 27, no. 9, pp. 2377–2388, 2016.

[4] N. Gupta, A. Prakash, and R. Tripathi, "Adaptive Beaconing in Mobility Aware Clustering Based MAC Protocol for Safety Message Dissemination in VANET," *Wireless Communications and Mobile Computing*, vol. 2017, pp. 1–15, 2017.

[5] X. Tang, D. Hong, and W. Chen, "Data Acquisition Based on Stable Matching of Bipartite Graph in Cooperative Vehicle–Infrastructure Systems," *Sensors*, vol. 17, no. 6, p. 1327, 2017.

[6] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, pp. 151–162, Seattle, DC, USA, August 1999.

[7] G. Korkmaz, E. Ekici, and F. Özgüner, "An efficient fully ad-hoc multi-hop broadcast protocol for inter-vehicular communication systems," in *Proceedings of the 2006 IEEE International Conference on Communications, ICC 2006*, pp. 423–428, tur, July 2006.

[8] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 84–94, 2007.

[9] X. Shen, X. Cheng, L. Yang, R. Zhang, and B. Jiao, "Data dissemination in VANETs: a scheduling approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2213–2223, 2014.

[10] W. Zhu, D. Gao, and C. H. Foh, "An Efficient Prediction-Based Data Forwarding Strategy in Vehicular Ad Hoc Network," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 128725, 2015.

[11] J. Zhu, M. Liu, Y. Wen, C. Ma, and B. Liu, "Parking backbone: toward efficient overlay routing in VANETs," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 291308, 13 pages, 2014.

[12] L. Zhang and B. Jin, "Dubhe: A reliable and low-latency data dissemination mechanism for VANETs," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 581821, 2013.

[13] A. Reyes, C. Barrado, M. López, and C. Excelente, "Vehicle density in VANET applications," *Journal of Ambient Intelligence and Smart Environments*, vol. 6, no. 4, pp. 469–481, 2014.

[14] R. S. Schwartz, H. Scholten, and P. Havinga, "A scalable data dissemination protocol for both highway and urban vehicular environments," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, article 257, 2013.

[15] K. Z. Ghafoor, K. A. Bakar, S. Salleh et al., "Fuzzy logic-assisted geographical routing over vehicular ad hoc networks," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 7, pp. 5095–5120, 2012.

[16] C.-J. Huang, Y.-W. Wang, H.-M. Chen et al., "An adaptive multimedia streaming dissemination system for vehicular networks," *Applied Soft Computing Journal*, vol. 13, no. 12, pp. 4508–4518, 2013.

[17] C. Wu, X. Chen, Y. Ji et al., "Packet size-aware broadcasting in VANETs with fuzzy logic and RL-based parameter adaptation," *IEEE Access*, vol. 3, pp. 2481–2491, 2015.

[18] U. Lee, S.-H. Lee, K.-W. Lee, and M. Gerla, "Understanding processing overheads of network coding-based content distribution in vanets," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 11, pp. 2304–2318, 2013.

[19] B. Hassanabadi and S. Valaee, "Reliable periodic safety message broadcasting in VANETs using network coding," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1284–1297, 2014.

[20] S. A. M. Ahmed, S. H. S. Ariffin, and N. Fisal, "Network coding techniques for VANET advertising applications," *Eurasip Journal on Wireless Communications and Networking*, vol. 2015, no. 1, article 200, 2015.

[21] C. Wu, X. Chen, Y. Ji, S. Ohzahata, and T. Kato, "Efficient broadcasting in VANETs using dynamic backbone and network coding," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6057–6071, 2015.

[22] C. Wu, S. Ohzahata, Y. Ji, and T. Kato, "Joint fuzzy relays and network-coding-based forwarding for multihop broadcasting in vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1415–1427, 2015.

[23] C. Wu, S. Ohzahata, and T. Kato, "VANET broadcast protocol based on fuzzy logic and lightweight retransmission mechanism," *IEICE Transactions on Communications*, vol. E95-B, no. 2, pp. 415–425, 2012.

[24] S. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, 2003.

[25] A. Ghazi Zadeh, A. Fahim, and M. El-Gindy, "Neural network and fuzzy logic applications to vehicle systems: literature survey," *International Journal of Vehicle Design*, vol. 18, no. 2, pp. 132–193, 1997.

[26] S. Jung, S. Park, and S. Lee, "Effect of MAC throughputs according to relative velocity in vehicle ad hoc network," in *Proceedings of the 2007 International Conference on Convergence Information Technology (ICCIT 2007)*, pp. 1183-1182, Gyeongbuk, Korea, November 2007.

[27] M. Bando, K. Hasebe, K. Nakanishi, and A. Nakayama, "Analysis of optimal velocity model with explicit delay," *Physical Review E: Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, vol. 58, no. 5, pp. 5429–5435, 1998.

[28] T. Tang, Y. Wang, X. Yang, and Y. Wu, "A new car-following model accounting for varying road condition," *Nonlinear Dynamics. An International Journal of Nonlinear Dynamics and Chaos in Engineering Systems*, vol. 70, no. 2, pp. 1397–1405, 2012.

[29] L. Fangxun, C. Rongjun, G. Hongxia, and L. Siuming, "An improved car-following model considering the influence of optimal velocity for leading vehicle," *Nonlinear Dynamics*, pp. 1–10, 2016.

[30] J. Yoo, B. S. C. Choi, and M. Gerla, "An opportunistic relay protocol for vehicular road-side access with fading channels," in *Proceedings of the 18th IEEE International Conference on Network Protocols, ICNP'10*, pp. 233–242, Kyoto, Japan, October 2010.

[31] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic Theory and Applications*, Prentice Hall, New Jersey, NJ, USA, 1995.

[32] A. Ishikawa, M. Amagasa, T. Shiga, G. Tomizawa, R. Tatsuta, and H. Mieno, "The max-min Delphi method and fuzzy Delphi method via fuzzy integration," *Fuzzy Sets and Systems*, vol. 55, no. 3, pp. 241–253, 1993.

[33] A. Keranen, J. Andott, and T. Karkkainen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques (Simutools '09)*, pp. 1–10, Rome, Italy, March 2009.

[34] F. Bai, . Narayanan Sadagopan, and A. Helmy, "IMPORTANT: a framework to systematically analyze the impact of mobility

on performance of routing protocols for adhoc networks," in *Proceedings of the IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 825–835, San Francisco, Calif, USA.

[35] K. Zhixin, H. Liangmin, T. Yehui, and Y. Jianqiang, "Plankton community structure and diversity in coral reefs area of Sanya Bay, Hainan Province, China," *Biodiversity Science*, vol. 19, no. 6, pp. 696–701, 2011.

[36] K. Kumar, A. Prakash, and R. Tripathi, "A spectrum handoff scheme for optimal network selection in nemo based cognitive radio vehicular networks," *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 6528457, 16 pages, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 3859836, 13 pages
https://doi.org/10.1155/2017/3859836

Research Article

An ARM-Compliant Architecture for User Privacy in Smart Cities: SMARTIE—Quality by Design in the IoT

V. Beltran, A. F. Skarmeta, and P. M. Ruiz

Department of Information and Communication Engineering, University of Murcia, Murcia, Spain

Correspondence should be addressed to V. Beltran; vicbelma@gmail.com

Received 26 April 2017; Accepted 6 August 2017; Published 10 September 2017

Academic Editor: Lampros Lambrinos

Much has been said about the benefits that the Internet of Things (IoT) will bring to citizens' life. Countless smart objects will be soon offering autonomous behavior in smart environments by sensing the physical world around us, collecting information about us, and taking proactive actions (many times without our consent) with the ultimate goal of improving our wellness. Without a strong guarantee on user privacy, the IoT may sound scary for many citizens. Indeed, the IoT-Architecture Reference Model (IoT-ARM) is a European effort for promoting IoT quality aspects such as security and privacy. This paper paves the way to the adoption of reference architectures by describing the application of the IoT-ARM within a European-funded project, SMARTIE. The SMARTIE architecture has been designed to empower citizens to take control of their IoT devices and privacy, while guaranteeing scalability for large deployments in smart cities.

1. Introduction

Millions of smart objects will be around us soon in what we call smart homes, smart buildings, and smart cities [1]. For citizens, smart environments will bring ubiquitous innovative services that will make their everyday life easier and improve their wellness and even their health. However, the ubiquitous and autonomous nature of Internet of Things (IoT) devices has made the debate on user privacy hotter than ever. These devices many times do not expose user interfaces for privacy configuration and collect and share user data without users being aware of this. The benefits of the IoT will not be maximized if citizens perceive their privacy in peril and hence neglect to take part of IoT services. Nevertheless, the risk of losing citizens' trust on the IoT is not seen by IoT verticals that focus on accomplishing their application-specific goals, leaving important quality aspects such as security undefined or poorly applied. Video cameras that are left open to online viewing, Internet-connected automobiles that are hacked on highways, and automatic unlock mechanisms for homes that grant unauthorized access [2] are just some examples. Having witnessed the harmful consequences of cyberattacks in the Internet, one can easily imagine the serious threats of smart cities with millions of connected IoT nodes; some of them controlling critical infrastructures for transport, security, and health. To meet the expectations on the IoT, it is imperative to address its challenges and maximize its benefits while reducing its risks. Common consensus on security and other quality aspects is needed in heterogenous and interconnected smart cities. In this regard, in order to promote quality aspects of IoT platforms, the European Union (EU) has invested efforts on several FP7-programme-funded projects in the last few years [3]. Notably, the IoT-Architecture (IoT-A) project started in 2010 to develop a Reference Architecture and finally released the IoT-Architectural Reference Model (IoT-ARM) [4] in 2012. The ultimate goal of IoT reference architectures is to consolidate an IoT ecosystem for engineers to work under the frame of well-stated quality-of-service aspects. Likewise, stakeholders can rely on reference architectures as a framework that guarantees the quality of compliant IoT platforms and enables their comparison.

This paper introduces the IoT-ARM to readers through a real use case, the generation of the platform developed by the SMARTIE EU-funded project. Given the large and interconnected documentation on the IoT-ARM, this paper is intended to help readers to understand the benefits of this Reference Architecture and how they can generate their IoT platforms based on its specifications. SMARTIE is an

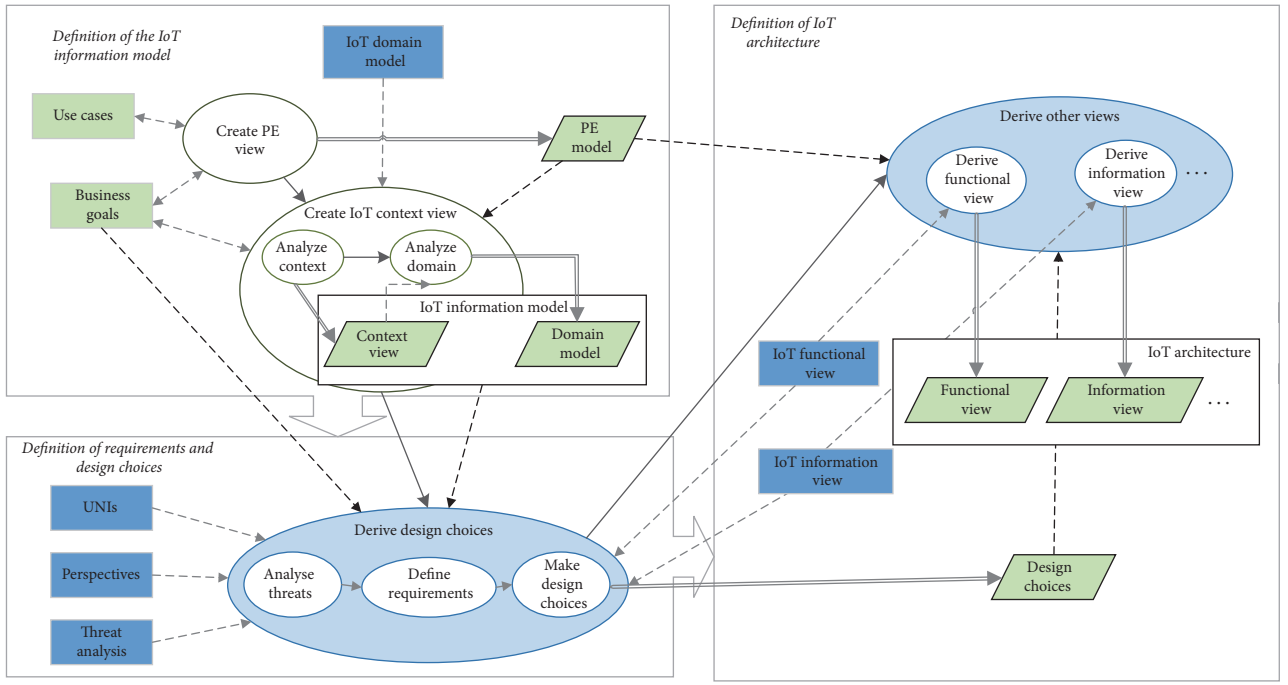


Figure 1: Main steps in the ARM-complaint architecture-generation process. Solid arrows indicate the flow of control, being "Create PE View" the starting process. Transparent circles are application-dependent processes and hence not specified by the IoT-ARM. Blue-colored circles are processes modeled by the IoT-ARM. Dashed arrows indicate inputs and outputs. Blue rectangles represent IoT-ARM inputs and green parallelograms the outputs of processes.

integrating IoT platform that supports the secure and efficient dissemination of IoT data in smart cities. Scalability and user privacy have therefore been the two major quality perspectives for the SMARTIE platform. Scalability is provided by the distribution and decentralization of most of the SMARTIE functionality. Privacy is guaranteed by Functional Components that allow citizens to be in control of the disclosure of their sensitive data: decentralized policy-based access control, encryption, and secure device bootstrapping.

This paper is organized as follows. Section 2 describes the IoT-ARM. Section 3 illustrates the ARM-compliant architecture-generation process through the SMARTIE platform. Section 4 introduces the SMARTIE architecture and describes the main components for user privacy. Section 5 describes the interaction between the main Functional Components of SMARTIE for a particular use case. Section 6 introduces some related work, and finally Section 7 gives some conclusions.

2. The Basics of the IoT-ARM

The IoT-ARM was conceived as an abstract and application-independent reference framework in order to support the generation process of IoT architectures in any IoT domain. Thus, the IoT-ARM defines high-level concepts, semantics, and functions that are common to any IoT platform. It is composed of two main blocks [4]: the IoT Reference Model and the IoT Reference Architecture. The former constitutes a general information model for the IoT that architects can use as the foundation for their application-specific information model. The latter serves as the basis and guidance for the design and derivation of concrete IoT architectures. The rationale behind the IoT-ARM is the development of IoT platforms that satisfy the stakeholders' concerns on quality aspects. To this end, IoT architectures are formed by architectural views that are designed to accomplish well-defined qualitative requirements. The IoT-ARM relies on perspectives to represent qualitative aspirations on (a) evolution and interoperability, (b) availability and resilience, (c) trust, security, and privacy, and (d) performance and scalability.

Figure 1 outlines the three main phases involved in the generation of an IoT-Architecture based on the IoT-ARM: the definition of the IoT Information Model, the definition of design choices, and lastly the definition of the IoT-Architecture. As it can be seen in the last phase, an IoT-Architecture is a composition of architectural views. Architects take the high-level architectural views defined in the IoT Reference Architecture as a reference to design their own views representing their particular application's requirements. The following subsections introduce the fundamental IoT-ARM blocks that support the three phases outlined in Figure 1.

2.1. The IoT Reference Model. The IoT Reference Model (from now on RM) provides a common understanding of the IoT domain for any IoT platform. The RM provides three basic submodels for the architecturing process: the IoT Domain Model, the IoT Information Model, and the IoT Functional Model. Architects rely on these three models to roughly

define the functionality and information flows that their IoT platform should provide.

The IoT Domain Model forms the basis for the rest of submodels by providing a common taxonomy of the main IoT concepts and their relationships [5]. This common taxonomy represents the backbone of the information model of any specific IoT domain. As depicted in Figure 1, the IoT Domain Model is the main IoT-ARM's input for the definition of any application's information model.

In the IoT Domain Model, there are seven core concepts: Physical Entity (PE), Virtual Entity (VE), Augmented Entity (AE), User, Device, Resource, and Service. A PE is any physical object that is relevant from a user or application perspective. An AE is a combination of a PE and its digital representation, that is, its VE. In a typical IoT scenario, VEs are associated with Resources that reflect the state of the related PEs (e.g., the temperature resource of a temperature sensor). Services can expose Resources and can be associated with VEs. Users can interact physically with PEs and digitally with Services. Indeed, a User can be either a person or a software agent.

The IoT Information Model gives more details about VEs (i.e., relations, attributes, and services) at a conceptual level. Thus, this model can be seen as an augmentation of the information provided by the IoT Domain Model.

The IoT Functional Model is an abstract framework for understanding the main Functionality Groups (FGs) of any IoT-Architecture and their interactions.

2.2. IoT-ARM Requirement Process. The IoT-ARM takes a quality by design approach by defining a requirement process as a previous step to the generation of any particular architecture. This process, which is depicted by the second phase of Figure 1, results in a set of choices for the design of the IoT-Architecture. It relies on three main sources of high-level requirements and design choices: Unified Requirements (UNIs), Perspectives, and Threat analysis. Architects can rely on these components to derive or instantiate their concrete design choices as follows.

2.2.1. Perspectives and Tactics. The IoT-ARM links qualitative perspectives to a set of abstract tactics that should be followed to accomplish the perspective's quality properties. Architects have to translate perspectives' tactics to concrete design choices for their system architecture. To support this critical task, the IoT-ARM gives a set of design choices for the architectural views impacted by perspectives' tactics.

2.2.2. Unified Requirements. The IoT-ARM defines a set of UNIs that are formulated on a high abstraction level in order to be applied to any potential domain-specific IoT system. Each UNI is associated with relevant information such as the driving (high-level) business goal, involved concepts of the IoT Domain Model, and the impacted components of the IoT Reference Architecture (i.e., architectural views and FGs). On one hand, architects can take UNIs as the basis to instantiate requirements for their particular needs. On the other hand, through UNIs' associations, architects can easily identify the components that are impacted by UNIs and hence should be especially considered. Moreover, some UNIs are associated with perspectives, thereby allowing architects to explore the quality aspects that should be addressed for these UNIs and the design choices that would help satisfying these UNIs.

2.2.3. Threat Analysis. The IoT-ARM provides a Threat analysis that assesses common risks for any IoT system. This analysis on one hand concludes a set of mitigating design choices and, on the other hand, can serve as an inspiration to define concrete requirements for any particular architecture.

2.3. The IoT Reference Architecture: Architectural Views. Architectural views represent system aspects that can be conceptually isolated, namely, the PE View, the IoT Context View, the Functional View, the Information View, and the Deployment and Operation View.

The PE View identifies the physical entities that will be central for the IoT system. The IoT Context View represents, on one hand, how the system interfaces to the outside world and, on the other hand, the domain model of the system. These two views are not defined by the IoT-ARM since they are use-case-dependent. The Functional View provides a set of Functional Components (FCs) for each FG identified in the IoT Functional Model. The Information View describes how information is handled and exposed by FCs as well as the information flows between them. The Deployment and Operation View is a set of guidelines to help architects to realize concrete systems based on their defined IoT-Architecture.

3. Generation of the SMARTIE Architecture

This section outlines how the SMARTIE platform's architecture has been developed based on the IoT-ARM. We introduce the four most important building blocks for generating an IoT-Architecture (see Figure 1) from our experience: the definition of business goals and use cases, the design of the domain model, the derivation of design choices, and the definition of the architecture's Functional View. We refer the reader to [4] for further information on the ARM-compliant architecture-generation process.

3.1. Business Goals and Use Cases. The definition of the IoT system's use cases and business goals is the starting point for architects. SMARTIE defines several uses cases for smart cities such as intelligent public transportation, traffic management, and energy management and safety in smart buildings [6]. The core business goal of SMARTIE is "*to enable the efficient and secure dissemination of data in smart cities based on a user-centric privacy- and security-by-design approach*." Moreover, the SMARTIE project has developed a set of smart city services integrated with the SMARTIE platform. For each of them, main business goals that represent functional goals were determined [6, 7]. Due to space limitations, only one of these services is considered: the smart management of emergencies and energy consumption in buildings. In this use case, business goals include the following: "*the system must be capable of detecting emergency events such as fire*

and notifying these events to authorized parties," "*emergency notifications should include information about the people that is within the building for quickly and effectively responding to the emergency*," "*the system must be capable of detecting the building places where there might be users and whether they have mobility restrictions*," and "*the system must be capable to efficiently reduce energy consumption of the building based on human presence.*"

3.2. Definition of the Architecture's Domain Model. The defined business goals and use cases are used to build the most elementary architectural view, that is, the PE View. In this view, we think about the things of interest and their properties. As stated in the IoT-ARM, a PE is an identifiable part of the physical environment that is of interest to the user for the completion of some goal. In our use case, the most evident physical entities of interest are the people that are within the building. One of the goals of the system is to help on rescuing people in case of emergency. We come back to the above definition of PE to highlight that PEs must be uniquely identified. In our application scenario, people will be provided with a unique Radio-Frequency Identification (RFID) card. Note that a PE can also be any entity that is part of the environment and is needed for a software artifact to complete some goal. It suggests that sensor devices that enable monitoring the presence of people and the status of the building will be PEs of our system too. Indeed, the IoT-ARM RM defines the Device class as a subclass of PE [4]. Thus, we have decided to consider these devices as PEs in our information model. Nevertheless, whether or not to consider devices as PEs is questionable and dependent on each specific application's design choices. In our system, the most evident example of IoT devices as PEs are RFID sensors that control the access of people holding RFID cards. Other useful devices are sensors that indicate when windows are open, when lights or Air Conditioning (ACs) systems are switched on/off, video cameras with human-detection capability, and so forth. Lastly, the IoT system should be able to monitor building spaces such as rooms and halls, thereby being these spaces PEs for our system too.

Based on the PE View and the defined business goals, the next step is to create our IoT Context View that represents the IoT system's information model. This view is composed by the context view and the domain model. Both models are complementary and essential for the rest of the architecting process. The context view describes the relationships, dependencies, and interactions between the system and its environment. The domain model provides a deep insight into the relationships between the system entities and also interactions with the outside world. Usually, it is easier to first build the context view and afterwards the domain model, since the level of detail given for outside interfaces is lower than that of the IoT system.

Figure 2 shows the SMARTIE context view for the building management use case. The Building Management Office provides RFID cards for access to the building. The Building Automation System (BAS) is a smart gateway (GW) between the building's physical world (i.e., sensors and actuators) and the SMARTIE platform. The building's security system is an external entity that provides additional safety information such as presence detection based on video-camera records. The SMARTIE platform will monitor the status of the building and detect locations with human presence. Based on this information, the Energy Management Service will optimize the energy consumption of the building by controlling lights, AC systems, and energy supplies from solar panels. In case of emergency, information about human presence in the building including personal information (e.g., reduced mobility conditions and telephone numbers) will be notified to Emergency Managers.

Figure 3 shows a subset of the SMARTIE domain model that includes the two pivotal concepts in any IoT domain: the PEs of interest and users. As stated in the IoT-ARM, a user is a human person or a software agent that needs to interact with a PE. Users can be either human beings or Active Digital Artifacts (ADAs). An ADA is a running software application, agent, or service that may access other services or Resources. By keeping this definition in mind, we define several users (upper right corner of Figure 3) such as Emergency Managers, Building Visitors, and Building Automation System (BAS). An Emergency Manager is a person that can be notified of emergency events. In case of emergency, this user may need to interact with the system to do some action or get information for safety (e.g., to know if there is some open door in case of fire). A Building Visitor is a person that is within the building and interacts physically with the environment (e.g., he opens a window and turns on an AC). A BAS is a software agent that interacts with the building's sensors and actuators. A BAS acts as smart gateway (GW) between the building and the platform.

The upper left corner of Figure 3 shows the SMARTIE platform's PEs. A Registered Person in Figure 3 represents people that have been previously registered and given an RFID card. Note that a Registered Person PE is a Building Visitor User too, since the former interacts with the building. We have defined classes for the kinds of sensors and actuators of interest for our application. PEs can be an aggregation of other PEs. Based on this property, we define the Smart Space class that represents a building space composed of sensors and actuators. Each PE is represented in the IoT system by a VE that can be an ADA or a Passive Digital Artifact (PDA) such as a database entry. We have defined high-level VEs such as the Smart Space VE that is a composition of other lower-level VEs such as the Light Sensor VE. Services are associated with VEs and expose Resources hosted on PEs. For example, the Emergency Detection Service informs about the state of the Smoke Detector resource (i.e., binary state of "smoke detection") that is hosted by the Smoke Detector PE.

3.3. Derivation of Requirements and Design Choices. Once we have formulated our business goals, PE View, and IoT Context View, we can proceed with the requirement engineering process to finally derive the design choices that will impact on the last process, that is, the derivation of architectural views. The ultimate goal of SMARTIE is to facilitate the integration of user-centric privacy and governance into IoT applications for smart cities. Thus, security and privacy are first-class business goals in order to enable citizens to (1) control their

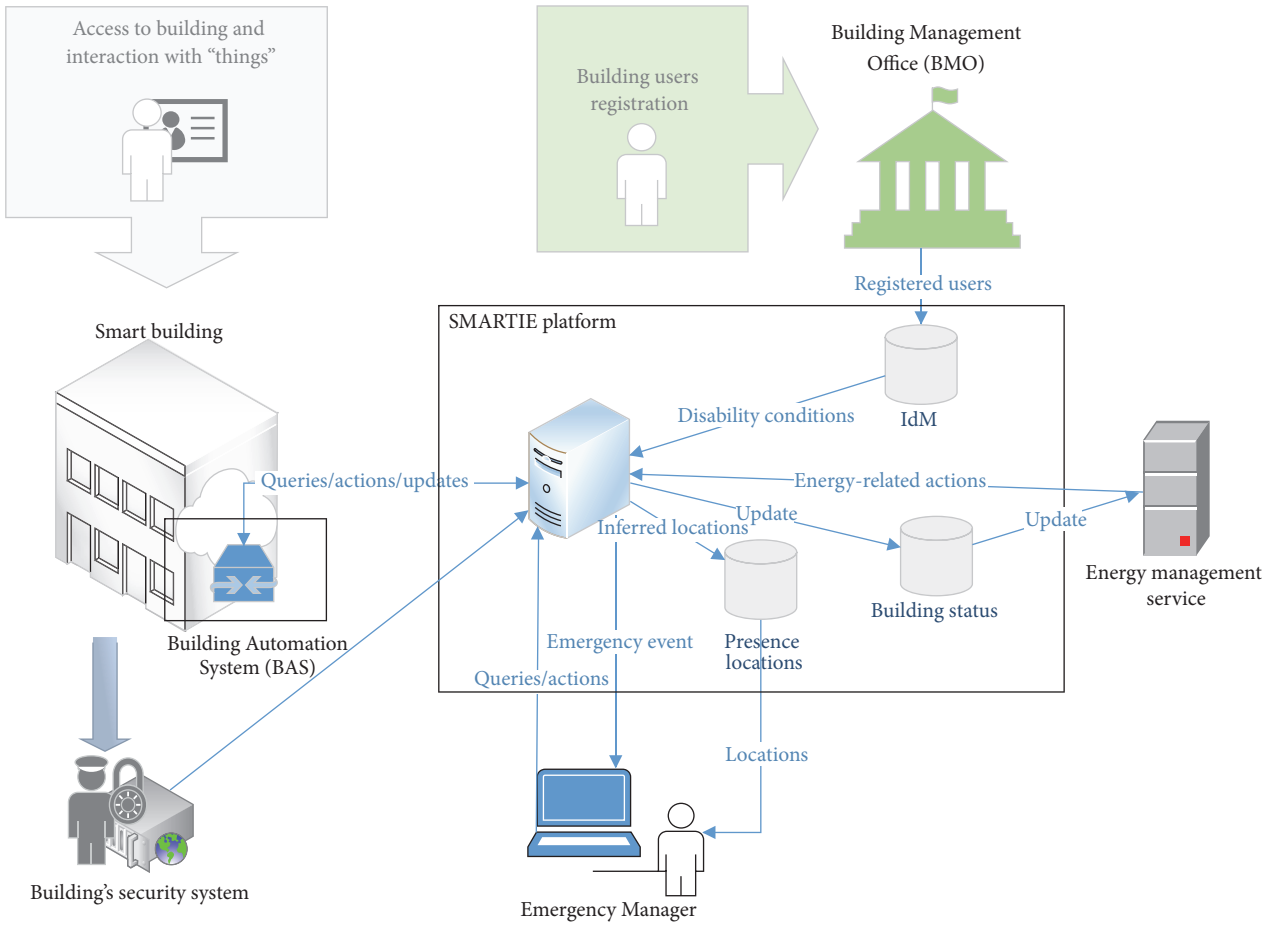


Figure 2: SMARTIE context view for the smart building use case.

devices that join an IoT application to sense and publish data, (2) define fine-grained access control rules for their devices, and (3) decide who can or cannot be in possession of their devices' data. Thus, major requirements and design choices of the SMARTIE platform were deduced from a deep analysis on the IoT-A Threat analysis, UNIs, and tactics. Table 1 shows only a few of SMARTIE design choices, mainly related to access control under different quality perspectives. Section 4 introduces the SMARTIE architecture's components for these design choices. More information about the requirements for the SMARTIE platform can be found at [7].

As it can be seen in Table 1, the IoT-ARM allows determining concrete design choices and correlate them based on different qualitative perspectives and tactics. The design choices taken for the Functional View will impact on the rest of the architecture's views. A relevant design choice about the SMARTIE's functionality was the enforcement of context-aware user access control policies by data producers such as sensors to improve user privacy (e.g., S-DC P.7 under the Privacy perspective in Table 1). Thus, only the actual authorized data will be granted by IoT devices rather than providing sensor data to centralized servers in charge of applying privacy filters. To accomplish this design choice while guaranteeing scalability, other choice on functionality was taken: decentralized access control for IoT devices (e.g., S-DC SP.4 and SP.6 under the scalability and performance perspective in Table 1). To facilitate the extensibility of the SMARTIE platform (e.g., integration with different IoT applications) and device-to-device communication, other design choice was to separate access control from application logic as much as possible (e.g., S-DC EI.1 and EI.2 under the evolution and interoperability perspective in Table 1).

To ensure user privacy, sensitive information needs to be end-to-end encrypted (DC S.10 under the security perspective). This decision requires other design choices at the Information View. For pull communication, SMARTIE uses transport-level encryption (S-DC EI.3 in Table 1). For push communication based on subscriptions, SMARTIE encrypts sensor data based on application-level user-defined attributes that data receivers must satisfy (e.g., S-DC S.5 in Table 1). This design choice ensures user privacy policies regardless of who receives the data (e.g., S-DC P.1 in Table 1); only those receivers that satisfy certain application-level attributes will be able to decipher the data.

3.4. Generation of the Architecture's Functional View. We took the IoT-A Functional View as the foundation for the SMARTIE's architecture. We modified this view based on

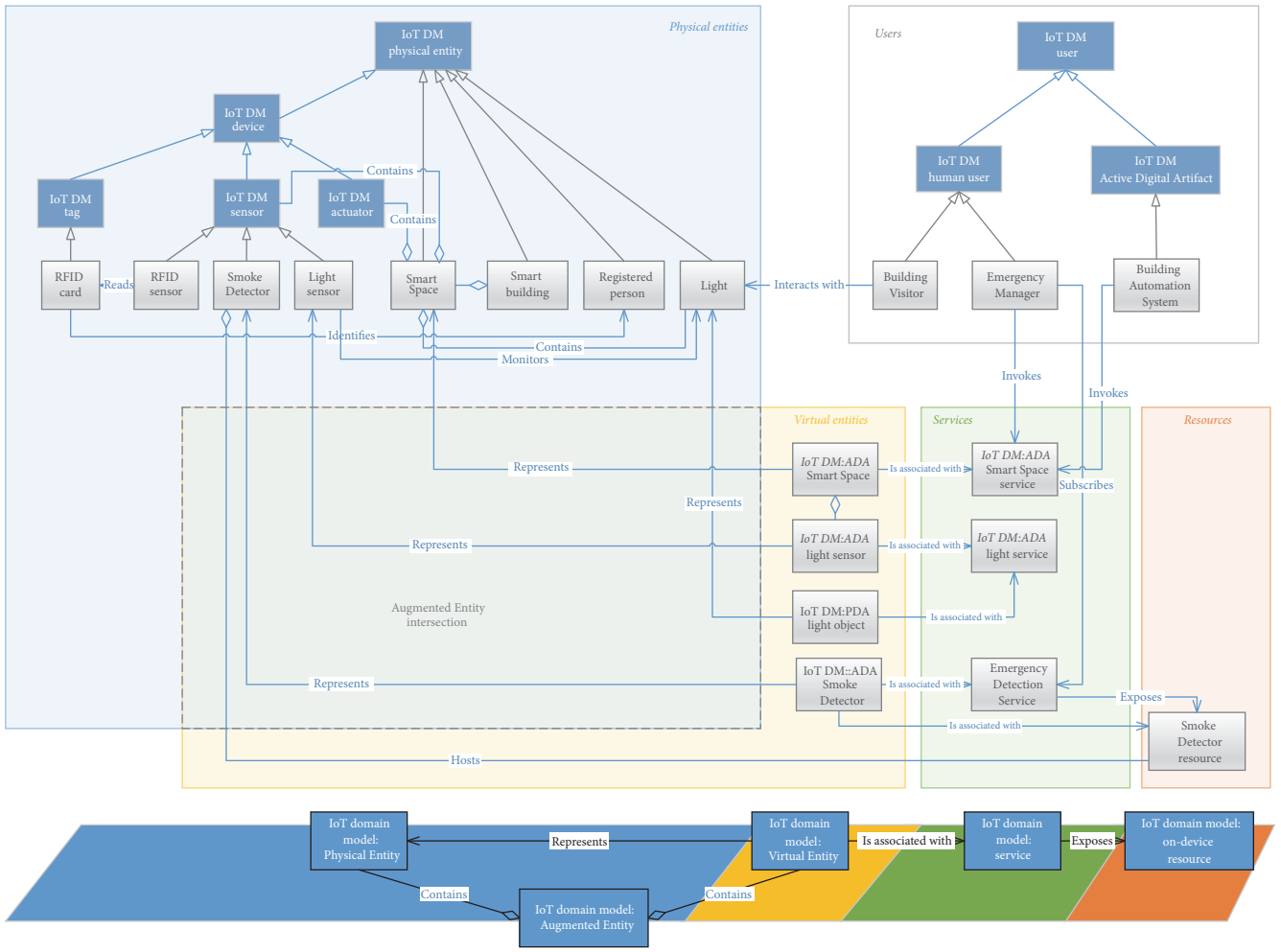


FIGURE 3: Subset of the SMARTIE domain model for the building management use case, and the main associations between the PE, VE, Service, and Resource concepts of the IoT Domain Model. The conceptual combination of a PE with its VE is an Augmented Entity (represented by dashed-lined box).

the design choices for SMARTIE, resulting in the functional architecture shown in Figure 4. In this architecture, the Communication FG represents the variety of communication technologies (e.g., data representation, addressing, and network management) that can be used by devices in IoT systems and provides a common interface for the IoT Service FG.

The Management and the Security FGs contain vertical functionality that can be used by any other FG in the architecture. The former provides all the functionalities that are necessary to govern an IoT system. The latter is responsible for ensuring security and privacy in the IoT system and is further described in Section 4.

The Service Organization FG is used for composing and orchestrating services of different levels of abstraction. The IoT Broker FC provides asynchronous communication (i.e., based on subscriptions/notifications) to match service requests with service offers. This FC relies on the VE Georesolution FC to find out the required IoT Services.

Applications can interact with the IoT system at the VE level that models high-level concepts of the physical world (e.g., "give me the status of windows in the room 102"). The VE Georesolution FC allows registering services by indicating the VE with which they can be associated and discovering services based on location information. The VE-Service FC provides access to VE Services (e.g., "switch off lights in room 102").

Besides the VE level, applications can interact with the IoT system at the IoT Service level by directly communicating with services hosted by devices (e.g., "give me your status" on a temperature sensor). Typically, IoT Services interact with devices and/or network Resources. High-level services are possible such as the Emergency Detection Service and the Energy Consumption Reasoner. The DiGcovery FC allows discovering IoT services by a service description or a service identifier, and it accepts location parameters to filter responses [8]. The Resource Directory with Secure Storage (RD) FC enables the automated registration of services and stores service information encrypted. The RD FC notifies the DiGcovery FC each time a new service is stored in the directory. In turn, the DiGcovery FC will automatically create

Table 1: IoT-A tactics for several perspectives and their associated IoT-A and SMARTIE design choices, identified by "DC" and "S-DC," respectively, for different architectural views. Design choices for SMARTIE are highlighted in italic. The design choices in the Functional View impact the design choices in the Information View and the Deployment and Operation View. In the last view, the design choices indicate the deployment's FCs that will be affected.

Perspectives	Tactics	Design choices (DC) for views		
		Functional	Information	Deployment and Operation
Security (S)	Use access policies	Policy-based service access (DC S.4)	Stored information must be managed to support access control mechanisms (DC S.6)	Authorization FC (DC S.7)
			Stored information can be encrypted based on application-level attributes (S-DC S.1)	*Key Exchange and Management FC (S-DC S.2) Deploy Authorization FC at end-devices (S-DC S.3)*
	Secure communication infrastructure	End-to-end encryption (DC S.10)	*Information transmission channel for devices or pull communication is secured (S-DC S.4)*	End-to-End, Network Communication and Key Exchange and Management FCs (DC S.12)
			Application data is encrypted for push communication (S-DC S.5)	*Authorization, DEM, Service Orchestration FCs (S-DC 6)*
Evolution and interoperability (EI)	Apply design techniques that facilitate change	*Access policies separated from application logic (S-DC EI.1)*	*Access policies must be in a standard, flexible format (XACML or JSON) (S-DC EI.3)*	*Authorization FC (S-DC EI.5)*
		Access control in devices must not require synchronization from centralized servers (S-DC EI.2)	*Authorization information is included in client requests (DCapBAC method) (S-DC EI.4)*	
Privacy (P)	Minimized unauthorized access to implicit information	Access control management (DC P.5)	Stored information managed to support access control mechanisms (DC P.6)	*Authorization, Key Exchange and Management FCs (S-DC P.2)*
			Data is encrypted based on application attributes (CPABE method) (S-DC P.1)	
		Enablement of a scalable and secure key distribution between communicating subjects (DC P.8)	*Constrained services dynamically learns authorized client's key based on authorization tokens (S-DC P.3)*	
	Enable the user to control their privacy settings	*Support flexible privacy rules (S-DC P.4)*	*Privacy rules provided by a Policy Decision Point with a flexible data format (S-DC P.5)*	*Authorization FC (S-DC P.6)*
		Enforce user's privacy settings through authorization rules at edge devices (S-DC P.7)	*Privacy rules translated to rules in authorization tokens for devices (S-DC P.8)*	

Table 1: Continued.

Perspectives	Tactics	Design choices (DC) for views		
		Functional	Information	Deployment and Operation
Scalability and performance (SP)	Reduce computation complexity	FC with reduced capabilities (DC SP.14)	No impact	Less functional component deployed (DC SP.15)
		Pushed information is not authorized (S-DC SP.1)	*Implicit authorization for pushed information based on attributes (CP-ABE) (S-DC SP.2)*	*Authorization, Service Orchestration and Key Exchange and Management FCs (S-DC SP.3)*
	Minimize the use of shared resources	*Authorization server does not synchronize access policies at devices (S-DC SP.4)*	No impact	*Authorization FC devices (S-DC SP.5)*
		Devices do not ask authorization servers for access control decisions (S-DC SP.6)	No impact	*Authorization FC devices (S-DC SP.7)*
	Optimized repeated processing	*Lightweight cryptographic operations for constrained devices (S-DC SP.8)*	*Cryptography based on lightweight methods (e.g. elliptic curve) for devices (S-DC SP.9)*	*Authentication, Key Exchange and Management FCs (S-DC SP.10)*

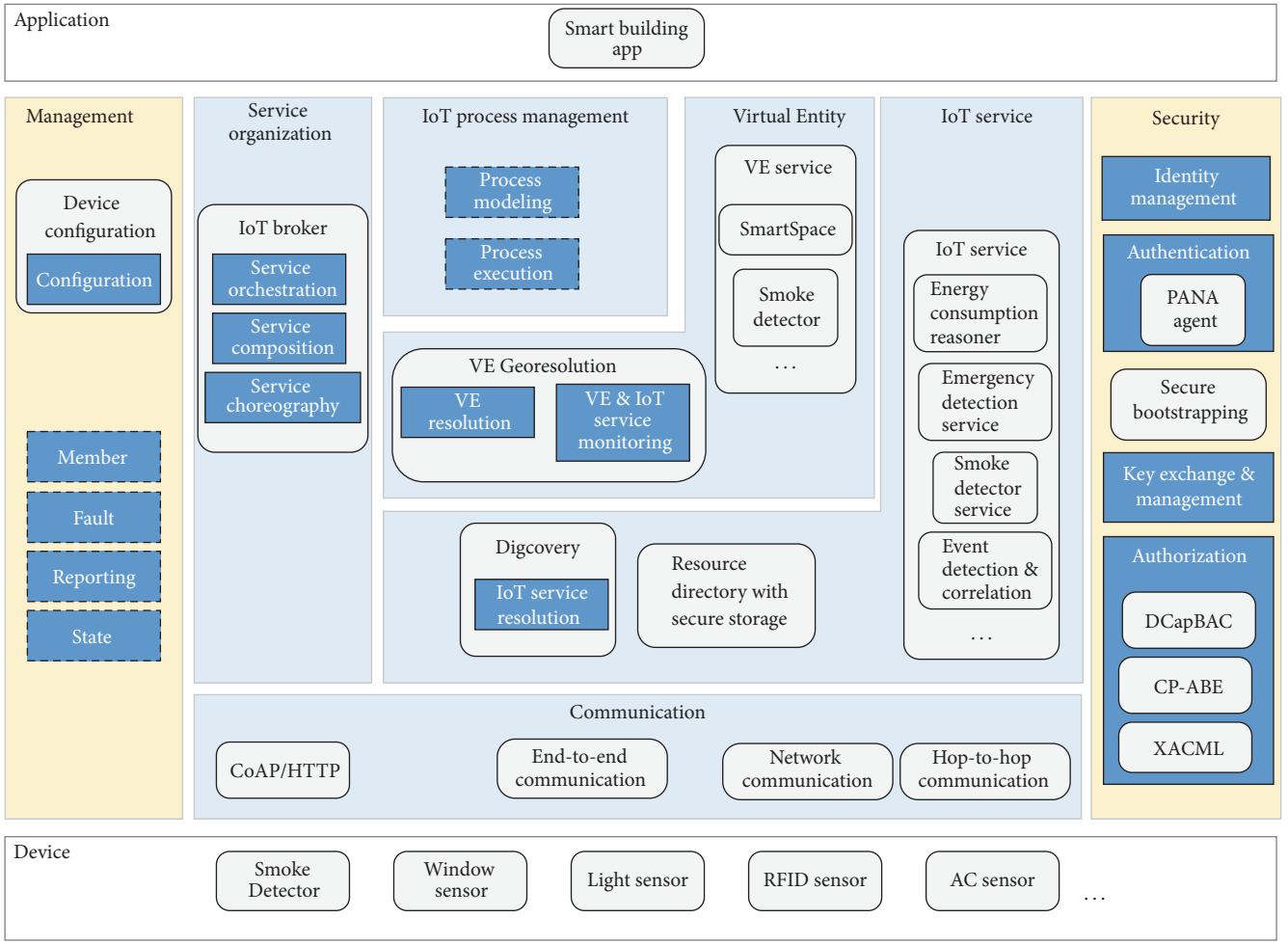


Figure 4: Simplified version of SMARTIE platform's Functional View and its Functional Groups (FGs). The device and application FGs are out of the scope of the ARM. Grey-colored rounded rectangles represent FCs defined for SMARTIE and dark blue-colored rectangles are FCs from IoT-ARM.

VE-Service associations in the VE Georesolution FC when new IoT Services are registered.

4. Functional Components for User-Centric Privacy

The Security FG of the IoT Functional View includes FCs for Authorization, Key Exchange and Management (KEM), Trust and Reputation, Identity Management (IdM), and Authentication. The SMARTIE architecture includes all these FCs except the Trust and Reputation FC, that is, planned to be considered in the near future. Figure 4 shows the primordial security-related FCs in the SMARTIE architecture, but the rest of FCs are described in [9].

The IoT Authorization FC is mainly distributed between three SMARTIE FCs: the eXtensive Access Control Markup Language (XACML), Decentralized Capability-Based Access Control (DCapBAC), and Ciphertext-Policy Attribute-Based Encryption (CP-ABE) FCs. These architectural artifacts satisfy many of the SMARTIE requirements and architectural design choices for user-centric governance, privacy, and scalability described in Section 3.3 and listed in Table 1.

The XACML FC as its name states allows defining fine-grained attribute-based access control policies through XACML. This FC stores and handles the user's context-aware authorization policies that will determine the devices that can join specific IoT applications and the entities that can access to these devices. Thus, any IoT application that allows users to define their access control policies by the XACML standard can be easily integrated with SMARTIE.

The DCapBAC FC is a delegated authorization mechanism [10] that flexibly allows a client device or application to access to a resource at a server device. This FC is distributed between the devices (i.e., the server and the client) and the SMARITE platform that provides authorization decisions. This authorizing SMARTIE functionality is called Authorization Server (AS). The client device requests the AS authorization to access to the server device. If this authorization request is granted, the AS generates a self-contained JSON-encoded authorization token that embeds lightweight but context-aware authorization rules. Authorization tokens enable the

server device to locally authenticate and authorize the client as long as they are signed by a recognized authority. The DCapBAC FC transforms user-defined XACML-based rules into lightweight JSON-based rules embedded in authorization tokens. Authorization tokens are integrity-protected and are resilient against centralized server failures (i.e., as long as the authorization token has not expired, clients and servers can directly communicate without any centralized authority). The DCapBAC FC therefore guarantees that user privacy policies are accomplished by any of the user's devices. This FC also facilitates the integration of user devices into multiple IoT applications since authorization rules follow a RESTful approach, without application-specific access control logic.

The CP-ABE FC implements an encryption-based authorization mechanism that encrypts data based on dynamic attribute-based policies that data consumers must hold [11]. This FC contains the policy attributes associated with data types and encrypt data based on these attributes. Here, a data type can be associated with a single data producer (e.g., "video from my vc identified by vc-entrance") or aggregated data (e.g., "my location" combines data from multiple sources such as the user's cell phone or laptop). When the IoT Broker FC needs to notify some information to a set of consumers, it requests the CP-ABE FC to encrypt this information by specifying the data type.

5. Real Scenarios for a Smart City

The SMARTIE IoT-ARM-compliant platform has been across country deployed for different use cases such as smart traffic management, public transportation, and environmental alarms [12]. The use case considered in this paper, emergency and energy management in smart buildings, has been tested in University of Murcia (UMU), Spain. The SMARTIE platform has been deployed for a smart building with 8 floors and a total area of 6.500 m^2. For the considered use cases, RFID sensors, video cameras and sensors for windows, doors, AC systems, and lights inform the Emergency Detection Service and Energy Consumption Reasoner about human activity in the building. The performance of SMARTIE in this scenario has been evaluated by taking time measurements of each of its components [13].

Figure 5 shows the main communication flows for the secure dissemination of data from sensor devices and some of the FCs of the SMARTIE architecture in Figure 4. The BAS connects to Home Automation Modules (HAMs) that serve as GWs to sensors and actuators. HAMs are intelligent modules that are distributed throughout the building and provide uniform interfaces to the BAS. The HAMs, BAS, and some devices can directly interact with the platform through RESTful communication (i.e., HTTP or CoAP).

When devices start up, they join the SMARTIE platform based on the Secure Bootstrapping FC that is composed of several other FCs (step 1 in Figure 5). A device first authenticates the PANA FCs that implements an extension of the Protocol for carrying Authentication and Network Access (PANA) [14]. This extension merges device authentication and authorization in order to save Resources at constrained devices [15]. The device needs to prove possession of a previously installed SMARTIE symmetric key and the PANA will query the XACML FC for the authorization of the device's key to join the platform. If this authorization request is successful, the PANA FC will register the device to the RD FC based on the privacy rules set for the device at the XACML FC (step 2 in Figure 5). Moreover, the PANA FC will reply to the device with the public key of the sensor's AS for future client requests. If the device was a sensor that had to periodically publish to the IoT Broker, the PANA would also reply to the device with an authorization token for publication (step 3 in Figure 5).

When the RD FC completes a device registration request, it notifies the DiGcovery FC that the device has been registered. In turn, the DiGcovery FC registers an association between the device's IoT Service and its corresponding VE to the platform's VE Georesolution FC.

When a sensor publishes to the IoT Broker FC (step 4 in Figure 5), the sensor attaches its authorization token to the body of its CoAP request towards the IoT Broker FC. Since the authorization token is signed by the DCapBAC FC, the IoT Broker FC can verify the authenticity of this token. Moreover, the sensor needs to be authenticated by proving possession of the public key contained in the token during the Datagram Transport Layer Security (DTLS) handshake with the IoT Broker FC.

When an application connects to the platform, it authenticates the Authentication FC that relies on the IdM FC (step 5 in Figure 5). After authentication, the KEM FC will provide the application with the proper attributes and cryptographic material to decrypt notifications from the IoT Broker FC. To this end, the KEM FC will communicate with the CP-ABE FC to obtain the necessary information. To subscribe to the IoT Broker, the application first obtains an authorization token from the DCapBAC FC and uses it to subscribe to a given service (steps 6 and 7 in Figure 5, resp.). The IoT Broker FC does not verify if the application is authorized to access to this required service's data. Instead, it creates the subscription and whenever the service produces some data, this FC requests the CP-ABE FC to encrypt this data and notifies all the subscribers of the encrypted data.

6. Related Work

The Alliance for Internet of Things Innovation (AIOTI) (http://www.aioti.org/) was launched by the European Commission in March 2015 in order to create and standardize an IoT ecosystem in Europe. This alliance is currently consolidating an IoT Reference Architecture mainly based on the results from IoT-A and oneM2M. The latter is an ETSI initiative that started developing their Reference Architecture [16] in parallel to the IoT-A project. There are other emerging initiatives that are intended to promote interoperability for large-scale IoT deployments. The IEEE Standard for an Architectural Framework for the Internet of Things (IEEE P2413) [17] defines a three-tier architectural framework, addressing descriptions, definitions, and common aspects in different IoT domains. The ITU-T Y.2060 "Overview of the Internet of Things" recommendation [18] follows a similar approach by providing a more harmonized view about the

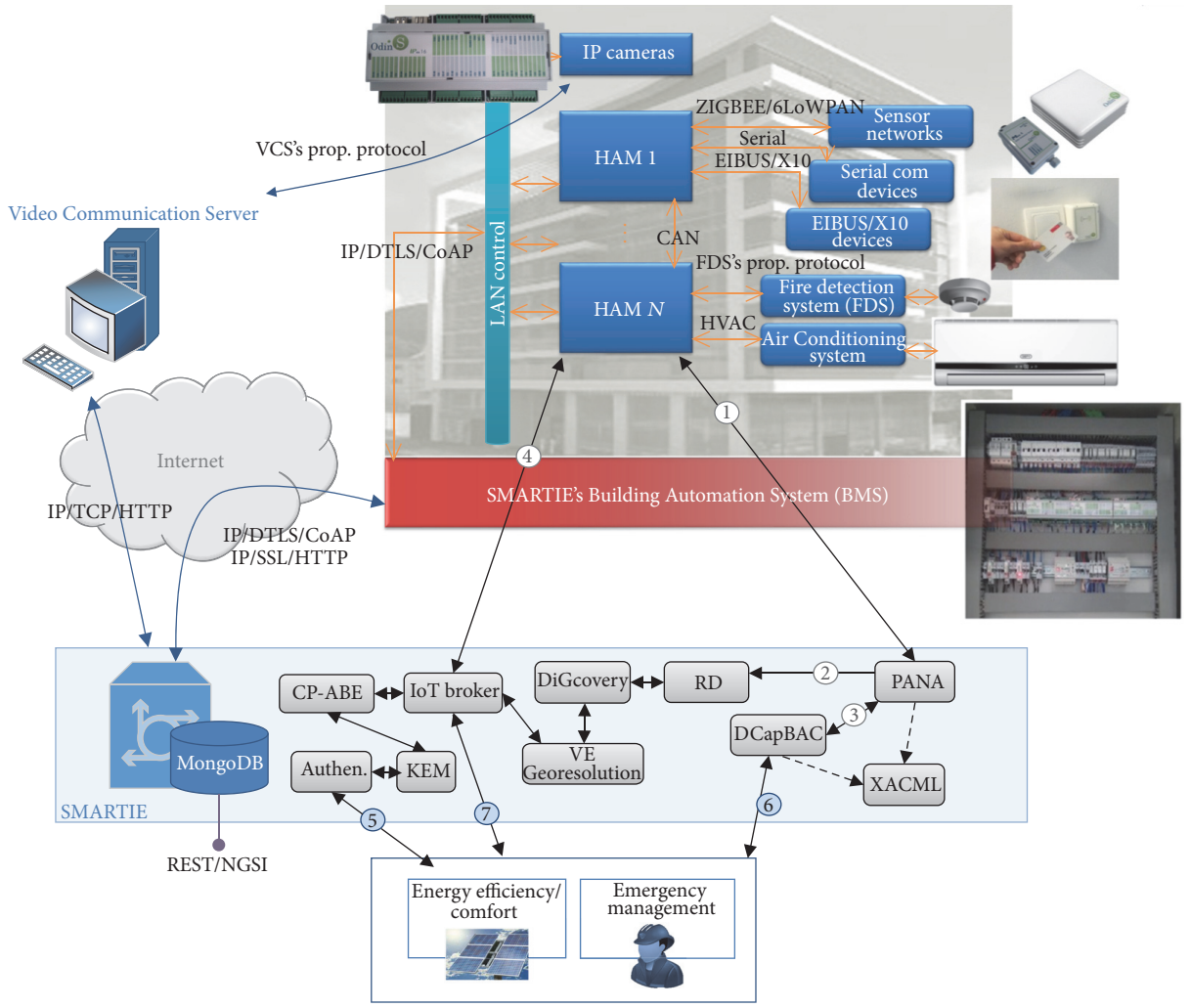


FIGURE 5: Outline of some Functional Components (FCs) of the SMARTIE platform for the building management use case. Arrows indicate flows of information between FCs. HAMs (picture at the top) are IoT gateways with μ-C-powered boards (32-bit MIPS μC running at 80 MHz, with 16 configurable inputs/outputs) that support RS-232, RS-485, ZigBee, 6LowPAN, and Bluetooth connections. Although arrows only depict the case when sensors are associated with HAMs, IP-enabled sensors to directly communicate with the BMS or the platform. Sensors (images at the top-right corner) have a 16-bit μC that runs at 8 MHz and support 802.15.4/6LowPAN/CoAP.

IoT ecosystem. IoT reference architectures are today a recent research topic and analyses that show the application of the IoT-ARM is scarce. The authors of [19] have to a very limited extent addressed this topic. However, this paper only shows some ARM-compliant architectural aspects of the proposed architecture.

SMARTIE has focused on (1) the by-design integration of security and privacy in IoT applications through the IoT-ARM and (2) the scalability of smart cities based on the decentralization of core functionalities for the secure dissemination of data. Other EU-funded projects have used the IoT-ARM for designing their platforms. The COSMOS project relied on the IoT-ARM to identify the requirements for its platform for decentralized management of things in the IoT [20]. This project provided a trust and reputation model based on different kinds of security threats. The FIESTA-IoT project aims to provide a common framework to access to and share IoT datasets in a testbed-agnostic way. The project provides an analysis of different IoT testbeds based on the IoT-ARM that is used as the foundation for its platform design [21]. Other EU-funded projects have also their focus on the dissemination of city data or the enhancement of the current state of IoT by integrating security and privacy. The City Platform as a Service (CPaaS.io) project is developing a platform for merging city data form a diversity of sources (e.g., social media, IoT data, and government data) and making this data available to third-parties (https://cpaas.bfh.ch/). The RERUM project [22] aims to make IoT more secure by providing a middleware based on OpenIoT [23]. It provides CoAP with JSON signatures based on Elliptic Curve Cryptosystem (ECC) of at least 192 bits for message integrity. SOCIOTAL [24] looks at IoT security from a societal point of view. Its main goals are user trust, user control, and transparency with the ultimate goal of obtaining the confidence of everyday

users and citizens. BUTLER project integrates OAuth 2.0 between authorization servers and clients for the obtention of access tokens and the derivation of security material [25].

Since the literature on IoT security is extensive, it falls out of the scope of this paper due to space limitations. We refer the reader to the references in this paper to find out more about the main aspects on security and privacy of SMARTIE: decentralized access control, encryption-based authorization, and secure bootstrapping.

7. Conclusion

This paper has given the authors' insights into the application of the IoT-ARM to generate the architecture of the SMARTIE, an IoT platform for secure and privacy-preserving dissemination of data in smart cities. The main goal of this platform is to empower citizens to take control of their privacy policies and devices. To this end, based on the IoT-ARM guidelines on security and scalability, SMARTIE provides architectural artifacts for efficient and scalable security and user-centric privacy. The paper has introduced SMARTIE user access control for pull communication (i.e., decentralized authorization tokens) and push communication (i.e., data encryption based on application attributes). SMARTIE provides an application-agnostic, scalable, and privacy-preserving platform for data dissemination in large deployments of smart cities.

One of the goals of the SMARTIE EU-funded project has been to evaluate the IoT-ARM for the generation of IoT platforms. Although the IoT-ARM represents a big step towards the homogenization of quality aspects in IoT platforms, further work on this Reference Architecture is necessary. In its current state, its steep learning curve may discourage some architects from using it.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been sponsored by European Commission through the FP7-SMARTIE-609062 EU Project and the Spanish National Project CICYT EDISON (TIN2014-52099-R) granted by the Ministry of Economy and Competitiveness of Spain (including ERDF support).

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] G. Ho, D. Leung, P. Mishra, A. Hosseini, D. Song, and D. Wagner, "Smart locks: Lessons for securing commodity internet of things devices," in *Proceedings of the 11th ACM Asia Conference on Computer and Communications Security, ASIA CCS 2016*, pp. 461–472, Xi'an, China, June 2016.

[3] S. Krco, B. Pokric, and F. Carrez, "Designing IoT architecture(s): a European perspective," in *Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT '14)*, pp. 79–84, Seoul, South Korea, March 2014.

[4] A. Bassi, M. Bauer, M. Fiedler et al., *Enabling Things to Talk: Designing IoT Solutions with the IoT Architectural Reference Model*, Springer Berlin Heidelberg, 2013.

[5] S. Haller, A. Serbanati, M. Bauer, and F. Carrez, "A domain model for the internet of things," in *Proceedings of the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, GreenCom-iThings-CPSCom 2013*, pp. 411–417, Beijing, China, August 2013.

[6] "SMARTIE Use Cases", SMARTIE project Deliverable 2.1, http://www.smartie-project.eu/download/D2.1-Use%20Cases.pdf.

[7] SMARTIE Requirements", SMARTIE project Deliverable 2.2, http://www.smartie-project.eu/download/D2.2-Requirements.pdf.

[8] A. J. Jara, P. Lopez, D. Fernandez, J. F. Castillo, M. A. Zamora, and A. F. Skarmeta, "Mobile digcovery: discovering and interacting with the world through the internet of things," *Personal and Ubiquitous Computing*, vol. 18, no. 2, pp. 323–338, 2014.

[9] SMARTIE Initial Architecture Specification", SMARTIE project Deliverable 2.3, http://www.smartie-project.eu/download/D2.3-Initial%20Specification.pdf.

[10] J. L. Hernández-Ramos, A. J. Jara, L. Marín, and A. F. Skarmeta Gómez, "DCapBAC: embedding authorization logic into smart things through ECC optimizations," *International Journal of Computer Mathematics*, vol. 93, no. 2, pp. 345–366, 2016.

[11] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the IEEE Symposium on Security and Privacy (SP '07)*, pp. 321–334, May 2007.

[12] "SMARTIE Real-world test- screenplay", SMARTIE project Deliverable 6.1, http://www.smartie-project.eu/download/D6.1-Real%20world%20test%20-%20Screenplay.pdf.

[13] "SMARTIE Test Report", SMARTIE project Deliverable 6.3, http://www.smartie-project.eu/download/D6.3-Test%20Report.pdf.

[14] D. Forsberg, Y. Ohba, B. Patil, H. Tschofenig, and A. Yegin, "Protocol for Carrying Authentication for Network Access (PANA)," RFC Editor RFC5191, 2008.

[15] D. Garcia-Carrillo and R. Marin-Lopez, "Lightweight CoAP-based bootstrapping service for the internet of things," *Sensors (Switzerland)*, vol. 16, no. 3, article no. 358, 2016.

[16] "oneM2M Functional architecture", TS 118 101 V.2.10.0, August 2016, http://www.onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_Architecture-V2_10_0.pdf.

[17] IEEE. Standard for an Architectural Framework for the Internet of Things (IoT) IEEE P2413. 2016. http://grouper.ieee.org/groups/2413/Intro-to-IEEE-P2413.pdf.

[18] ITU-T. Recommendation Y.2060 Overview of the Internet of Things. 2012. https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-Y.2060-201206-I!!PDF-E&type=items.

[19] J. Fernandes, M. Nati, N. S. Loumis et al., "IoT Lab: Towards co-design and IoT solution testing using the crowd," in *Proceedings of the 2015 International Conference on Recent Advances in Internet of Things, RIoT 2015*, Singapore, April 2015.

[20] S. Citrigno, S. Graziano, and D. Saccà, "Cooperation of Smart Objects and Urban Operators for Smart City Applications," in *Management of Cyber Physical Objects in the Future Internet of Things*, Internet of Things, pp. 157–174, Springer International Publishing, 2016.

[21] "Analysis of IoT platforms and Testbeds", Federated Interoperable Semantic IoT/Cloud Testbed and Applications (FIESTA-IoT) project deliverable D2.2, http://fiesta-iot.eu/wp-content/uploads/2016/06/FIESTAIoT-WP2-D22-web.pdf.

[22] G. Moldovan, E. Z. Tragos, A. Fragkiadakis, H. C. Pöhls, and D. Calvo, "An IoT middleware for enhanced security and privacy: The RERUM approach," in *Proceedings of the 8th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2016*, Larnaca, Cyprus, November 2016.

[23] J. Kim and J.-W. Lee, "OpenIoT: An open service framework for the Internet of Things," in *Proceedings of the 2014 IEEE World Forum on Internet of Things, WF-IoT 2014*, pp. 89–93, Seoul, South Korea, March 2014.

[24] Hernandez-Ramos, J. L., Bernal, J., Skarmeta, A., Elicegui I., Nati, M. Gligoric, N., WP2 – Decentralised governance and trust framework. http://sociotal.eu/sites/default/files/docs/deliverables/SOCIOTAL_D2.2-Framework_specification_for_Privacy_and_Access_Control_Final.pdf.

[25] S. U. Khan, L. Lavagno, C. Pastrone, and M. A. Spirito, "Online authentication and key establishment scheme for heterogeneous sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 718286, 11 pages, 2014.

Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 5414930, 18 pages
https://doi.org/10.1155/2017/5414930

Research Article

A Real-Time Taxicab Recommendation System Using Big Trajectories Data

Pengpeng Chen, Hongjin Lv, Shouwan Gao, Qiang Niu, and Shixiong Xia

School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China

Correspondence should be addressed to Shixiong Xia; xiasx@cumt.edu.cn

Received 8 January 2017; Revised 24 May 2017; Accepted 12 June 2017; Published 25 July 2017

Academic Editor: Paolo Bellavista

Carpooling is becoming a more and more significant traffic choice, because it can provide additional service options, ease traffic congestion, and reduce total vehicle exhaust emissions. Although some recommendation systems have proposed taxicab carpooling services recently, they cannot fully utilize and understand the known information and essence of carpooling. This study proposes a novel recommendation algorithm, which provides either a vacant or an occupied taxicab in response to a passenger's request, called VOT. VOT recommends the closest vacant taxicab to passengers. Otherwise, VOT infers destinations of occupied taxicabs by similarity comparison and clustering algorithms and then recommends the occupied taxicab heading to a close destination to passengers. Using an efficient large data-processing framework, Spark, we greatly improve the efficiency of large data processing. This study evaluates VOT with a real-world dataset that contains 14747 taxicabs' GPS data. Results show that the ratio of range (between forecasted and actual destinations) of less than 900 M can reach 90.29%. The total mileage to deliver all passengers is significantly reduced (47.84% on average). Specifically, the reduced total mileage of nonrush hours outperforms other systems by 35%. VOT and others have similar performances in actual detour ratio, even better in rush hours.

1. Introduction

Urban air and soil quality are essential to the health of urban residents. Good urban air and soil quality can greatly improve the function of the nervous system, enhance the efficiency of work, and ensure the healthy status of urban residents [1]. However, taxicab exhaust emissions have an extremely negative effect on urban soil [2] and air quality [3]. In Beijing, a taxi can run hundreds of thousands of kilometers a year [4, 5]. Under normal circumstances, exhaust emission from a taxi is more than 5 times the emission from a private car.

Carpooling services can effectively reduce the excessive emissions from taxis by reducing the total mileage to deliver all passengers. But unlike regular taxicab services that arbitrarily assign one vacant taxicab to a new passenger [6, 7], taxicab carpooling services require catching a particular taxicab, which refers to a taxicab with existing passengers heading to a direction similar to that of the new passenger. However, the occupied taxicab could not be found for a passenger based on the existing solutions for finding a vacant taxicab.

For the carpool service, there are mainly two categories: static and dynamic carpooling. In the static carpooling research, most researches focus on how passengers with similar destinations are assigned to a car [8–10] and how to improve the timeliness for the real-time performance of the carpooling service [11–13]. In all, the static carpooling problem in a sense can be regarded as a special member of the general class of the Dial-a-Ride Problem (DARP) [14].

Although the static carpooling researches have greatly improved the performance of carpooling services, the above researches are all built on the premise that the information of all passengers is known in advance. But the travel routes and time of existing passengers in taxicabs are not accessible for us on the basis of the existing infrastructure, unless we spend a huge fortune building a new thorough taxi system. In addition, the size of increasing vehicle data goes far beyond the range of DARP. Since the general DARP is NP-hard [15], only small datasets can be dealt with optimally [16, 17]. However, the further development of big-data-processing technology and the upgrading of taxi equipment (GPS [18] and fare meters), forming a huge GPS records database

with rich semantic information, provide an opportunity for predicting existing passengers' information, namely, the core of dynamic carpooling.

This paper belongs to the dynamic carpooling research. In dynamic carpooling, we do not have any information about travel routes and travel time of passengers in advance. What is more, reasonable request matching needs to be timely and efficiently accomplished with continuous query requests generated in real time. Thus, dynamic carpooling has the characteristics of real time, quick response, reasonable matching, and so forth. These characteristics are undoubtedly quite suitable for the large-scale taxi scene and more in accord with the needs of the public. Thus, this paper focuses on real-time dynamic carpooling based on taxicabs' GPS records.

Based on big-data-processing technology and historical taxicab GPS data, some researches [19, 20] provide a dynamic real-time carpooling service. However, existing dynamic carpooling researches have four defective aspects: (I) inadequate information mining, (II) ignoring valuable situations, (III) ignoring destination distribution characteristics, and (IV) one-sidedness of screening criteria. In Section 2, we will elaborate on these defective aspects and propose our motivations.

In this study, we propose *VOT*, a taxicab recommendation system based on extremely large taxicab GPS data. By using a unified standard to distinguish taxicab performances, *VOT* provides both carpooling and conventional taxicab services, which can effectively reduce the excessive emissions of taxis. The key contributions of this study are as follows:

(i) To the best of our knowledge, we propose the first carpool service, which can significantly reduce the total mileage to deliver all passengers under the premise of fully ensuring the interests of passengers. In addition, for raw GPS datasets with unstructured format, Spark is applied to improve the efficiency of large data processing.

(ii) To achieve our goal, we design a novel method to predict the occupied taxicabs' destination by similarity comparison and clustering algorithms. It can obtain more accurate forecasting destinations by fully mining GPS datasets and eliminating interferences from worthless destinations.

(iii) To more comprehensively evaluate the taxicab carpooling performance, we further propose a novel metric called Distance Dispersion, which is defined as an average distance between a particular passenger's destination and possible destinations of occupied taxicabs.

(iv) We evaluate *VOT* with a real-world dataset, containing 14747 taxicabs' GPS data. The results show that the ratio of range (between forecasted and actual destinations) of less than 900 M can reach 90.29% and *VOT* can reduce 53% of the total mileage to deliver all passengers, especially outperforming other systems by almost 35% at 0:00 to 7:00 AM.

The rest of the paper is organized as follows. Section 2 introduces our motivation. Section 3 presents taxicab networks research. Section 4 proposes our system overview. Section 5 depicts the system implementation. Section 6 validates our design with datasets. Several practical issues are discussed in Section 7, followed by the conclusion in Section 8.

2. Motivation

In this section, we present our motivations to improve the four legacy defects for taxicab carpooling services based on empirical data from a real-world taxicab network of 14747 taxicabs in Shenzhen [21].

First, we demonstrate theoretically four defects in the existing dynamic carpooling system and then further clearly interpret these deficiencies by figures and experiments. Finally, we discuss the methods we adopt to make up for these weaknesses.

2.1. Inadequate Information Mining. In dynamic carpooling services, we need to predict the potential destinations of these real-time occupied taxicabs for detecting this one with the best carpooling performance. However, we argue that although the potential destinations would be obtained eventually, little information (only the origin of occupied taxicabs and real-time passengers) is used to predict destinations in existing dynamic carpooling research. In other words, the potential destinations are inferred by finding similar trajectories that start at the same origin (the real-time occupied taxi) and pass the same location (starting point of passenger P) in other researches.

As shown in Figure 1(a), the passenger P sends a carpool request to the server at O_P. At this point, the real-time occupied taxi T (taking the existing passenger on O_T, passing through L1, L2, L3, ..., L9 to an unknown destination) in L9 can serve as a potential carpooling option for the passenger P, so we need to infer the destination of T (or the existing passenger) for quantifying its carpool performance.

As shown in Figure 1(b), existing dynamic carpooling studies use only C1 (the nearest intersection from T's origin O_T) and C4 (the nearest intersection from P's origin O_P) as the matching criteria. This approach ignores the valuable information between O_T and O_P. To the best of our knowledge, the more detailed the matching data we provide is, the more accurate our matching results will be. Therefore, the method of applying the last manned trajectory (between O_T and O_P) of T as the matching data in *VOT* is a necessary supplement to higher forecasting precision.

2.2. Ignoring Valuable Situations. The application of only two origins (real-time occupied taxicabs and passengers) not only results in the incomplete mining of the GPS dataset, but also ignores a great deal of valuable historical trajectory information (with high similarity).

Compared with the last manned trajectory of real-time taxicabs, the historical manned trajectories, especially with a higher degree of similarity, have a higher likelihood of having the same destination as these real-time taxicabs. Because

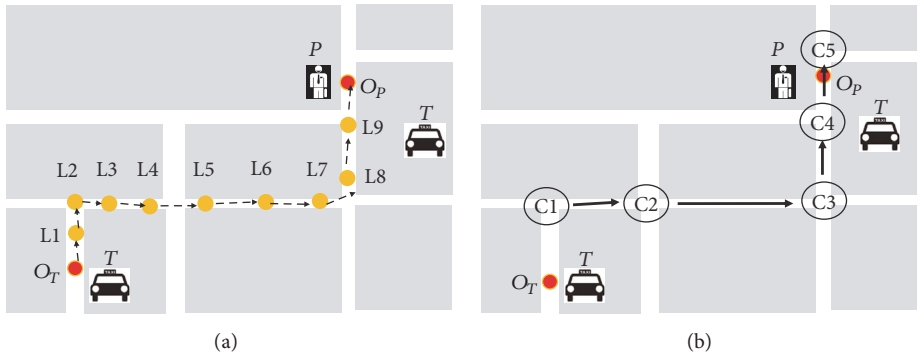


Figure 1: Real GPS records (VS) existing methods for dynamic carpooling.

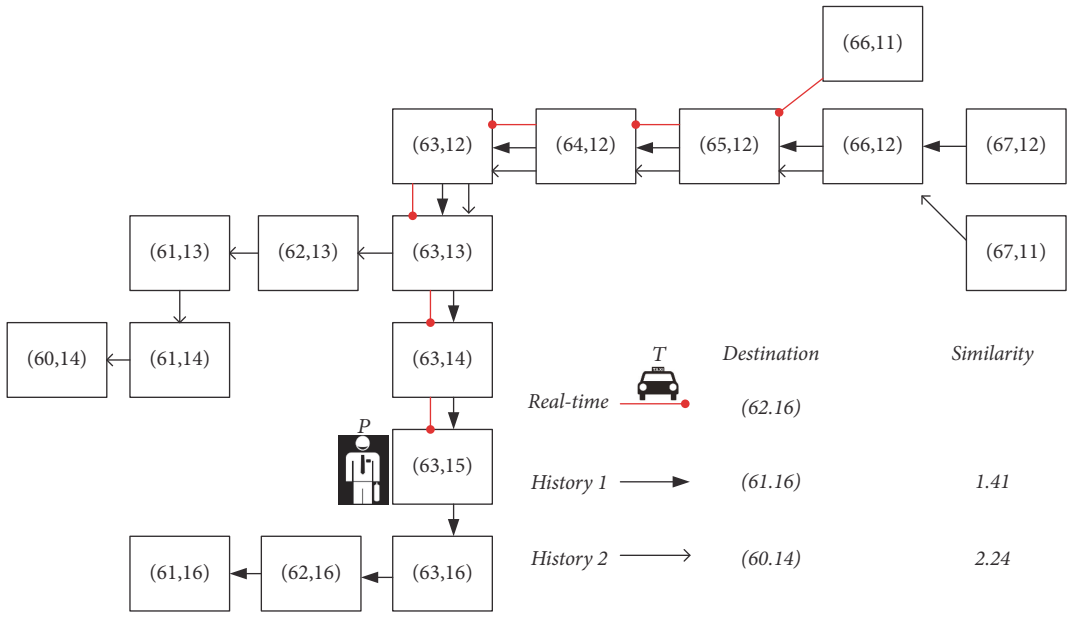


Figure 2: Ignoring valuable situations.

two pairs of latitude and longitude with the same value hardly exist, we introduce the regional division in this paper. Then, the map is divided into many marked regions (two-dimensional value, like (63, 15)).

We extract a case from experiments applying real GPS records and draw it up in Figure 2. As shown in Figure 2, when a passenger P asks for a carpooling request in (63.15) region, there is a real-time occupied taxicab T that can be regarded as a potential carpool option. Then, *VOT* puts the last manned trajectory data of T as matching data and compares it with the historical manned trajectory dataset. Compared with "History 2," the destination of "History 1" with greater similarity is closer to the destination of T. This confirms our previous conclusions.

2.3. Ignoring Destination Distribution Characteristics. After obtaining the initial potential destinations collection, existing dynamic carpooling schemes equally treat all destinations that appear in this collection and regard the frequency of potential destinations as their probability. However, there are two drawbacks if we follow the existing methods:

(A) Existing researches ignore the fact that the historical trajectories that generate the preliminary potential destinations collection have different similarity. In other words, each possible destination corresponds to different possibilities (by quantifying the similarity of historical trajectories). Compared with existing studies, this paper aims to detect those destinations with high similarity and high frequency, instead of only focusing on frequency.

(B) In the existing dynamic carpooling, a large proportion is allocated to massive possible destinations with quite low frequency. It has little chance to be the real destination when the region has few frequencies. Moreover, existing studies ignore the characteristics of destination distribution with regional distribution [22–24]. In other words, the vast majority of destinations are distributed in several hot spots.

Considering the above-mentioned limitations, we concentrate our efforts on finding the most likely regions and try our best to eliminate the interference of loose and extremely

low frequency destinations. Therefore, *VOT* makes use of clustering algorithms to divide the potential destinations and applies the cluster center to represent all the possible destinations in the same cluster. In this way, it highlights these regions with high frequency and high similarity to the greatest extent. What is more, even if the true destinations of real-time occupied taxicabs are not the forecasted cluster centers, the distance between these true destinations and cluster centers is quite small. To validate our design, we propose a new parameter in Section 6, called Real Prophecy Distance (RPD), to test *VOT* on the entire GPS dataset.

2.4. One-Sidedness of Screening Criteria. In this work, we argue that although carpooling choice would be obtained eventually, the ultimate carpooling choice should not be obtained by a parameter that can only meet the requirements of carpooling service in one side. Taxicab GPS records have been used by several systems to provide dynamic carpooling services. But existing researches, which mainly focus on detour distance, cannot perform well in both the interests of passengers and the mitigation of gas exhaust emissions.

As well known, if carpooling passengers have the same destination as the existing passengers, they would debus at the same time and place. Under this scenario, the carpooling service achieves its best utility, in which the carpooling passengers have no detour distance. Meanwhile, it reduces the mileage of the whole trip of carpooling passengers. In other words, a greater degree of closeness between the carpooling passengers' destination and the destinations of occupied taxicabs indicates lower extra consumption and a better carpooling performance.

Thus, we conduct our first work to provide carpooling services, which applies a novel parameter called Distance Dispersion to quantify the closeness between the destinations of particular passengers P and occupied taxicabs. The ultimate carpooling strategy for P in this paper is to select an occupied taxicab with the minimum Distance Dispersion as the "can-carpool" taxicab. In order to prove the superiority of Distance Dispersion, we evaluate the performance of *VOT* through actual detour ratio (%) and reduced total mileage (%) in Section 6.

3. Taxicab Networks Infrastructure

In this section, we present the taxicab networks infrastructure and the implicit semantic information inferred from the raw large GPS dataset.

3.1. Infrastructure. Underlying taxicab infrastructures in large cities are presently equipped with GPS, communication devices, and dispatch centers. Based on the upgrades of taxicab devices, the taxicab network can be roughly divided into two parts, namely, (1) numerous taxicabs, in the frontend, which provide service and assume the role of the sensing terminal at the same time, and (2) dispatching centers with cloud servers, in the backend, to receive and store sensing records for the taxicab service [25, 26].

The establishment of the large taxi GPS dataset is the foundation of system implementation. Based on the popularity of taxicabs' underlying infrastructure, these locations and statuses are periodically uploaded to the dispatching center, which forms a large taxi GPS dataset. The formation step of this dataset is presented as follows:

(1) Loaded with a wireless transmission module, the taxicab would cyclically send its status to the nearest cell tower.

(2) The status data would be forwarded to the cloud server by the cell tower.

(3) The real-time GPS data are stored in the cloud server established for analysis according to the fixed format.

Each GPS record of the large GPS dataset contains all the attribute categories of the taxicab real-time information. A GPS record mainly consists of the following parameters: plate number, which is the unique identification of taxicabs; date and time, which demonstrate the time of this record generated by the GPS device; GPS coordinates, which monitor the global status of the taxicab; Status Bit, which indicates if some passengers exist when this record is uploaded.

Real-time GPS records of tens of thousands of taxicabs would be uninterruptedly transmitted to the cloud server, forming large amounts of GPS trajectory information. Such raw large GPS dataset has a very high resolution, which can be used to locate a particular taxicab at fine granularity related to both time and space. Nonetheless, such a fine-granular large GPS dataset has many erroneous and missing records. Meanwhile, such a raw GPS dataset could not be obtained firsthand as it is in a format that is not ready for analysis [27]. In the next subsection, we extract useful implicit semantic information about the taxicab service from the raw large dataset.

3.2. Implicit Information in Underlying Infrastructure. Based on historical and real-time GPS records, we observe four statuses related to passenger demand by continuously tracking the GPS records of the same taxi.

(1) *Take-In Status.* For the same taxicab, if its status value turns from "0" to "1" in two consecutive records, then this taxi just picked up a passenger. The location of Take-In Status is considered an origin or a take-in location of a trip.

(2) *Drop-Off Status.* If the status value turns from "1" to "0" in two successive records, then this taxicab just dropped off a passenger. The location of Drop-Off Status is considered a destination of a trip.

(3) *Occupied Status.* Continuously observing the same taxi, if the status value keeps "1," then the taxi is heading to the destination of the passengers. We believe the location of Occupied Status is the middle section of one trajectory.

(4) *Wander Status.* When we continuously observe the GPS records of taxicabs, the taxi is at Wander Status if the status value holds "0" all the time.

Based on the implicit semantic information mined from the real-time GPS dataset, the regular taxicab recommendation systems can efficiently locate and recommend vacant

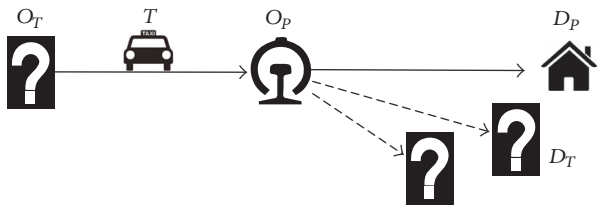


Figure 3: Semantic demonstration.

taxicabs to real-time particular passengers. Some existing recommendation systems even could provide a carpooling option when no nearby vacant taxicab is available. But they fail to guarantee result accuracy because of the low utilization of the large dataset and the inference from numerous worthless destinations with low frequency and low similarity. What is more, existing dynamics carpooling researches ignore the characteristics of destination distribution with regional distribution and cannot perform well in both the interests of passengers and the mitigation of gas exhaust emissions and traffic congestion.

Our extensive understanding on the large GPS dataset and carpooling service provides an opportunity to obtain higher inference accuracy. Based on the above analysis and discussion, our recommendation system locates and recommends the best taxicab in the performance of carpooling and conventional service to the real-time passenger, which is presented in the next section.

4. System Overview

This recommendation system is designed to mine GPS records in depth for enhanced recommendation quality. Considering that regular services are commonly understood, we provide a scenario in which carpooling services are applied, and then we present the main idea of our recommendation system.

4.1. Scenario Demonstration. Figure 3 presents a scenario in which passenger P requests for a taxi at origin (O_P) heading to destination (D_P). Built on the implicit semantic information in underlying taxicab infrastructure and specific passenger information, no taxis in the Wander Status are found around the passenger P. But, based on the observation on real-time GPS records, the recommendation system can locate nearby occupied taxi T as a potential "can-carpool" taxicab (heading to an unknown destination) that will pass the origin of P soon. Owing to the limited knowledge on the destinations of existing passengers on taxicab T, carpool service could not be reached just with the request of passenger P.

By reverse tracking on the real-time GPS records based on time, *VOT* obtains the last manned trajectory (between O_T and O_P) of T. Compared with this last manned trajectory, the historical trips, especially with a higher degree of similarity, have a higher likelihood of having the same destination as the existing passengers. Thus, *VOT* fully mines the historical and real-time GPS records and regards the destinations of highly similar historical trajectories as potential destinations.

VOT further optimizes the potential destination sets by the clustering algorithm catching center regions, which can efficiently summarize the features of destination distribution and thoroughly reduce the interference from worthless destinations with low frequency and similarity. In this study, we catch these center regions by using different clustering algorithms (K-means [28, 29], density-based spatial clustering of applications with noise (DBSCAN) [30, 31], and balanced iterative reducing and clustering using hierarchies (BIRCH) [32, 33]).

When a nearby occupied taxicab provides a carpooling service to the particular passenger P, the real trip of P generates additional consumption compared with the conventional taxi service. Therefore, the optimal carpooling strategy means a "can-carpool" taxi with the lowest consumption. A greater degree of closeness between the destinations of carpooling passengers and occupied taxicabs indicates lower consumption and a better carpooling performance.

Therefore, a novel parameter called Distance Dispersion is used to quantify the degree of closeness in *VOT*. Distance Dispersion could be obtained by averaging the Manhattan and the Euclidean Distances between the real-time passengers' destination and forecasted potential destinations. Different occupied taxicabs have different destinations, resulting in different Distance Dispersions for P to carpool. The optimal carpooling strategy for P is to select an occupied taxicab with the least Distance Dispersion as the "can-carpool" taxicab.

4.2. Main Procedure. The main procedure of *VOT* is presented in Figure 4.

4.2.1. Manned Trajectory Distributions. The taxicab manned trajectory distribution, which is the foundation of carpooling service, plays a crucial role in our recommendation system.

We separate individual trips from the entire historical GPS dataset by continuously tracking and observing the change in Status Bit on the GPS records of the same taxicab. The distribution, generated from the large GPS dataset, contains historical GPS records for all taxicabs. With the context of a particular passenger, such a distribution can generate the potential destinations of trajectories with a high degree of similarity compared to another certain trajectory.

4.2.2. Distance Dispersion Calculation. Based on the manned trajectory distribution, when receiving a request from passenger P, our recommendation system would apply the similarity comparison and clustering algorithm to calculate an expected Distance Dispersion ρ_T^P for P to carpool with a particular nearby taxicab T according to six different calculation models. All calculation models are divided into the following four steps:

(1) All systems first locate a nearby taxicab set T, where taxicabs are near the origin, based on the traces of taxicabs in the dataset for a particular day.

(2) According to the manned trajectory distribution and passenger P information, we can calculate a preliminary potential destination set MD_T^P for taxicab T.

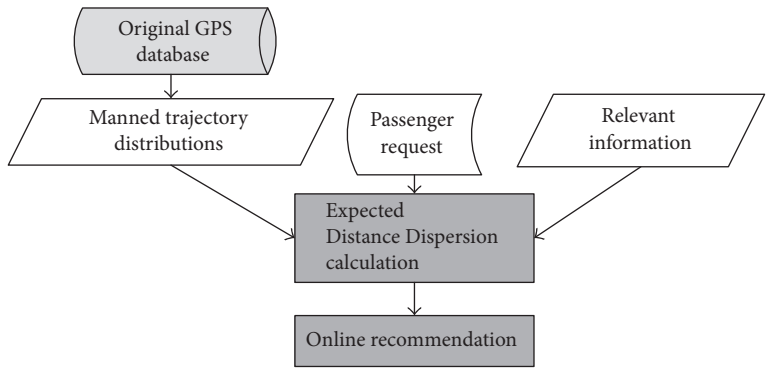


Figure 4: Main procedure.

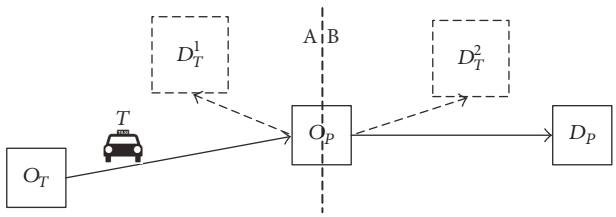


Figure 5: Basic model.

(3) Based on the context information, our system optimizes MD_T^P by removing worthless destinations to achieve a compact size in basic and advanced models. We then calculate RD_T^P containing the representatives of all potential destinations by the clustering algorithms for a further optimization.

(4) On the basis of RD_T^P, the recommendation system assigns probabilities and calculates the Distance Dispersion of this particular occupied taxi.

Basic K-Means

(1) When we receive a request, this scheme can calculate set T, where taxicabs are all near the request origin based on the real-time GPS records.

(2) By calculating the similarity between historical trajectories and the last manned trajectory of T, our system obtains the set MD_T^P, in which every potential destination has two attributes (frequency and average similarity).

(3) In the basic design, if a destination is the polar opposite of a passenger destination, then our recommendation system would eliminate this destination, due to that large consumption compared with conventional taxi service. As shown in Figure 5, when the possible destination D_T^2 of T is in B, D_T^2 is a closer destination to D_P, which diminishes consumption compared with D_T^1 in A.

K-means is then used to deeply optimize and highly generalize the characteristics of the taxi destination distribution.

(4) In the basic design, with assigning equal probabilities for the destinations in RD_T^P, *VOT* calculates a weighted average ρ_T^P by their locations.

Advanced K-Means. Advanced K-means is similar to *basic K-means* except for two differences.

In (3), the advanced design is built upon the basic design. However, in the advanced design, based on richer underlying information, our system further reduces the size of MD_T^P by two steps of depth optimization.

Step 1. We firstly census a set, called Recent Occur Destinations (ROD), which contains the destinations and their frequencies that have occurred in the recent days according to historical manned trajectories. And there are some potential destinations that do not appear in ROD or have only minimal frequencies (less than three times). Therefore, since these destinations have a small probability of being the real destination, *VOT* in the advanced model removes these destinations that have rarely occurred in recent days to improve prediction accuracy.

Step 2. If a region appears many times in a short period of time, this indicates that there has been a great service demand for this region in the last few hours. In other words, this region has a great possibility of being the real destination. Therefore, *VOT* firstly censuses these regions, which are the final destinations for manned trajectories that have occurred in recent hours. Then, *VOT* in the advanced model detects and marks the region with high frequency. At the end of clustering algorithms, we can obtain the middle region of the marked region and the cluster center in which this marked region is located. At last, the intermediate region replaces the original cluster center as the representative. These measures in Step 2 not only effectively solve the problem of short-term carpooling request surge caused by unexpected emergencies, but also compensate for the omission of real-time emergencies in Step 1.

In (4), after obtaining the clustering result from K-means, the recommendation scheme assigns probabilities to different representatives based on their individual frequencies, resulting in an accurate calculation in the Distance Dispersion. In

other words, the visits of these representatives are used as the basis for assigning probability. For example, if 10 trips starting from O_T exist in the distribution, four of them have D_T^x as their destination, whereas the others have D_T^y; our system then assigns $\Pr(D_T^x) = 4/10$ and $\Pr(D_T^y) = 6/10$ to calculate a weighted average ρ_T^P.

Basic and *advanced K-means* optimize MD_T^P by the K-means algorithm, a typical clustering algorithm based on distance. K-means uses distance as the similarity evaluation index; thus, the closer the two objects are, the greater the similarity is. The function method to find the extremum is used for the adjustment rules of iterative operation [28, 29]. The entire process is calculated as

$$\sum_{i=1}^{K}\sum_{j=1}^{N}\left(\left(F_j - F_i\right)^2 + \left(S_j - S_i\right)^2\right)^{1/2}, \tag{1}$$

where K is the number of initial cluster centers and N is the number of remaining destinations. F represents frequency, and S denotes average similarity.

The Minkowski Distance formula between two regions and the cluster center coordinate are shown below:

$$\mathrm{MK} = \left(\sum_{k=1}^{n}\left(x_{1k} - x_{2k}\right)^p\right)^{1/p} \left(\sum_{i=1}^{K+N}\frac{F_i^2}{K+N}, \sum_{i=1}^{K+N}\frac{S_i^2}{K+N}\right). \tag{2}$$

When $p = 1$, the Minkowski Distance is the Manhattan Distance; when $p = 2$, the Minkowski Distance is the Euclidean Distance.

Basic and *advanced DBSCAN* are similar to *basic* and *advanced K-means*, but they use DSBSCAN to optimize MD_T^P. DBSCAN is a spatial clustering algorithm based on density, which is not sensitive to distance. The algorithm divides the regions with sufficient density into clusters and finds the clusters of arbitrary shapes in noisy spatial databases [30, 31]. Based on the above reasons, *advanced DBSCAN* has the best performance in both Distance Dispersion and reduced total mileage on average, except when the density is uneven and the distance between clusters is very different at some time, which can also be proved in Section 6.

Basic and *advanced BIRCH* are also similar to *basic* and *advanced K-means*, but they use BIRCH to optimize MD_T^P. BIRCH is a clustering algorithm based on hierarchy [32]. This algorithm uses two concepts, namely, clustering feature and clustering feature tree, to generalize clustering description [33].

4.2.3. Online Recommendation. The algorithm recommends a real-time taxi with the minimum expected Distance Dispersion for a particular passenger by analyzing the Distance Dispersion for every nearby taxi whether in the Wander or in the Occupied Status.

5. System Implementation

5.1. Calculation Framework. Although the raw GPS dataset is typically of a large volume and interconnects multidimensional records with high resolution, much of the raw dataset is of no interest in our design. We need to map this raw physical GPS dataset to a filtered and compressed logical dataset for analysis. Moreover, we should process this raw physical GPS dataset by an intelligent method in order to meet the high timeliness and low latency requirements. In this aspect, a large data-processing framework can be a good solution to the problem of raw and massive data processing.

Spark [34] is the latest generation of software framework for distributed processing of large-scale data, which has the advantages of high efficiency, high fault tolerance, and low cost [35]. Memory distribution dataset goes into operation in Spark, which improves the performance of iterative computation by caching data in memory [36]. Thus, Spark meets the requirements of the real-time taxi recommendation system for high timeliness and low latency [37]. In conclusion, our recommendation system uses Spark to deal with the raw GPS dataset.

As a burgeoning big-data-processing model, Spark provides the basic abstraction that is a resilient distributed dataset (RDD [38]). RDD represents an immutable, partitioned collection of elements that can be operated in parallel. Data manipulation in Spark programs can be divided into three steps: the creation of RDD, the transformation of the existing RDD, and the operation of the RDD returning the computing result. In detail, before submitting the Spark program, Spark runs the program's main function and builds a Spark context. Then, Spark programs load data by abstracting data into a RDD. Finally, based on the user-defined logic, the data processing and transformation are realized on the basis of user-defined functions and the operator (map, filter, groupByKey, sortByKey, etc.) provided by Spark.

However, although the types of operators provided by Spark are rich, there are still some complex and unique operation logics, which need to be implemented by the combination with user-defined functions and the operators provided by Spark.

5.2. Historical Manned Trajectory Distribution. Each GPS record has a pair of latitude and longitude, but if the GPS latitude and longitude point are regarded as a mark of matching the trajectories, we could not map the particular trajectories because two pairs of latitude and longitude with the same value hardly exist. Therefore, we introduce regional division in *VOT*. The map is divided into many marked regions. A marked region would contain several GPS records of the same taxicabs by continuously tracking the GPS records. The taxicab trajectory can then be represented by a series of marked regions. In this manner, trajectory matching becomes possible by searching for particularly same regions.

Based on the raw large GPS dataset and regional division, *VOT* could obtain the manned trajectory distribution, in which each manned trajectory consists of a series of marked regions instead of one-by-one GPS latitude and longitude point to describe the entire taxi-manned trajectory. As shown

Table 1: Original GPS records and areas of manned trajectory.

Number	Time	Longitude	Latitude	Area	Status
23953	19:32:45 PM	114.0993	22.5451	Jd43Wd7	0
23953	19:32:49 PM	114.0989	22.5518	Jd43Wd7	1
23953	19:33:08 PM	114.0990	22.5401	Jd43Wd6	1
23953	19:33:26 PM	114.0988	22.5391	Jd43Wd6	1
⋮	⋮	⋮	⋮		⋮
23953	19:47:55 PM	114.0489	22.5321	Jd35Wd5	1
23953	19:48:01 PM	114.0479	22.5316	Jd35Wd5	1
23953	19:48:10 PM	114.0429	22.5312	Jd34Wd5	1
23953	19:51:20 PM	114.0409	22.5298	Jd34Wd5	0

(1) Input taxicabs GPS data after cleaning
(2) Using *map* transformation, the format of raw taxicabs GPS records is converted to (plate number, (date and time; marked region; status bit))
(3) Using *groupByKey* transformation, all the taxicabs GPS data of the same plate number are gathered.
(4) **if** (Using *filter* transformation, we inspect and detect if there are real-time taxicabs in Wander Status)
{
(1) Using *map* transformation, we calculate the distance between these real-time taxicabs and *P*. Then, (corresponding distance, plate number) are exported.
(2) Ascending order of corresponding distance can be obtained by *sortByKey(true)* transformation.
(3) Using *take(1)* operation, we obtain and recommend the nearest taxicab in Wander Status to the passenger *P*.
}
(5) **else**
{
(1) Using *filter* transformation, we inspect and detect if there are real-time taxicabs in Occupied Status.
(2) Using *map* transformation, GPS data for each taxicab are arranged in reverse chronological order. Then, we output plate number and the corresponding last manned trajectory, namely, several continuous GPS records in which the status bit is 1.
}

Procedure 1: Access to real-time taxicab information.

in Table 1, serval original GPS records are used as examples to demonstrate the above conversion.

The original GPS records are transformed to several marked regions (e.g., Jd43Wd6) after the regional division in Table 1. A series of raw GPS records describe the details of the above entire trajectory, which can be mapped on a given region map, corresponding to a unique carpool graph. Thus, a manned trajectory is extracted from raw GPS records and represented by a series of marked regions.

5.3. Function Implementation. The procedure of Spark data processing is a series of RDD transformations and operations. Hence, a series of key RDD transformations and operations are used to explain the critical details of the mechanisms and algorithms in this section. In the following, the significant RDD transformations and operations are described.

5.3.1. Access to Real-Time Taxicab Information. Upon receiving a request from passenger P in O_P, we first need to search for taxicabs around passenger P through the real-time GPS records. In Procedure 1, the real-time taxicabs in Wander Status or Occupied Status are obtained by testing the time and Status Bit of GPS records.

If there are several real-time taxicabs in Wander Status around P, the distances between O_P and these taxicabs are regarded as an attribute of vacant taxicab performance. Then, we select the nearest vacant taxicab to P. If there is no real-time vacant taxicab around P but only a few real-time

Step 1. Obtaining Initial Destination Sets
(1) Input the historical taxicab manned trajectory data. Then, the storage level of these trajectory data is put into *StorageLevel.MEMORY_ONLY* by *cache* method due to the need for repeated comparisons.
Input the last manned trajectory of real-time taxicabs in Occupied Status. *textFile* method is used to load the HDFS file into Spark as an initial RDD.
(2) Using *map* transformation, we can obtain the similarity between the last manned trajectory of these taxicabs and the historical manned trajectory data.
After the above operations, the new RDD with the format of (similarity, destination) is transformed.
(3) Using *sortByKey (false)* transformation, the descending order about similarity of potential destinations is obtained.
(4) Using *take(n)* operation, we can obtain n taxicab historical manned trajectories which have higher similarity, and destinations of these manned trajectories are regarded as a preliminary set MD_T^P.
(5) In order to deal with these data more conveniently and quickly, we change the form of MD_T^P to (destination, similarity) by *map* transformation. After that, the new MD_T^P is exported to HDFS to facilitate filtering operations later.

Step 2. Forecast Final Destinations
(1) *textFile* method loads and abstracts MD_T^P into RDD, and then *VOT* gathers the similarity of the same potential destination by *groupByKey* transformation.
(2) Using multiple operators provided by Spark and user-defined functions, downsized and optimized MD_T^P is obtained in basic and advanced models.
(3) Using *foreach* operation, we calculate the visit frequency and the average similarity of potential destinations in MD_T^P and export the data in the format of (potential destinations, (frequency, average similarity)) to HDFS.
(4) The new MD_T^P in HDFS is abstracted as RDD by the *textFile* method. Then, through a series of transformations and actions including user-defined functions, we implement and complete three different types of clustering algorithms and output the representatives RD_T^P of MD_T^P.
The format of initial RD_T^P is ((destinations and these attributes in Cluster A), (destinations and these attributes in Cluster B)...)
(5) Based on *map* transformation and initial RD_T^P, cluster centers and total visit frequency of clusters are calculated by user-defined functions.
The format of the output file is ((the cluster center C_A and total visit frequency N_A of Cluster A),...).
(6) We traverse each element of the RDD by the *foreach* operation to count the total visit frequency N. Then, the ultimate RD_T^P with the format of $((C_A, N_A/N), \ldots)$ is exported to HDFS.

Procedure 2

taxicabs in Occupied Status, we further calculate the last manned trajectory of the real-time taxicabs in Occupied Status (see Procedure 1).

5.3.2. Potential Destinations of Occupied Taxicabs. In Procedure 2, in order to obtain potential destinations of real-time taxicabs in Occupied Status, our algorithm is roughly divided into two steps.

Step 1 (obtaining initial destination sets MD_T^P). By the comparison between the last manned trajectory of these real-time occupied taxicabs and the historical manned trajectory data, *VOT* calculates and acquires the destinations of n trajectories which have higher similarity, namely, MD_T^P.

Step 2 (forecast final destinations RD_T^P). Based on the frequency and average similarity of every potential destination in MD_T^P, different clustering algorithms (K-means, density-based spatial clustering of applications with noise (DBSCAN), and balanced iterative reducing and clustering using hierarchies (BIRCH)) complete clustering operations. Then, we calculate and regard the cluster centers set RD_T^P as the representative of potential destinations in the same cluster.

5.3.3. Distance Dispersion Calculation and Optimal Recommendation. In order to screen out the real-time occupied taxicab with the best carpooling performance, Procedure 3 is divided into two steps.

Step 1. Our algorithm calculates the Distance Dispersion of every real-time taxicab in Occupied Status.

Step 2. This real-time taxicab in Occupied Status with the best carpooling performance is selected by *VOT* and recommended to the particular passenger P.

As shown in Procedure 3, our recommendation strategy specifies a *map* transformation that puts the representatives RD_T^P and P's request (origin and destination) as input file to calculate Distance Dispersion of real-time occupied taxicabs. The generic calculation formulas are as follows:

$$\rho_T^P = \sum_{D_T \in RD_T^P} \Pr(D_T) \left(\frac{\mathrm{EM}_{D_T}^{D_P} + \mathrm{MH}_{D_T}^{D_P}}{2} \right)$$

$$\mathrm{EM} = \left(\sum_{k=1}^{n} (x_{1k} - x_{2k})^2 \right)^{1/2} \quad (3)$$

$$\mathrm{MH} = \sum_{k=1}^{N} |x_{1k} - x_{2k}|,$$

(1) Input and abstract clustering results RD_T^P to new RDD.
(2) Using *map* transformation, the carpool performance of a single potential destination of a real-time taxicab in Occupied Status is quantified (Distance Dispersion) by user-defined functions.
(3) Using *reduceByKey* transformation, the carpool performance of a single real-time occupied taxicab is obtained by aggregating all Distance Dispersion of potential destinations in RD_T^P.
(4) By the *sortByKey* transformation, the descending order of the carpool performance (the ascending order of Distance Dispersion) is processed.
(5) Using *take(1)* operation, we obtain and recommend the real-time taxicab in Occupied Status with the best carpooling performance to the passenger P.

Procedure 3: Distance Dispersion calculation and optimal recommendation.

where RD_T^P is the representative of MD_T^P and D_T is a representative of the potential destinations. $\mathrm{EM}_{D_T}^{D_P}$ is the Euclidean Distance between passenger P's destination D_P and the real-time taxicab T's destination D_T. $\mathrm{MH}_{D_T}^{D_P}$ is the Manhattan Distance between these destinations. Every destination has a different probability according to the frequency by which it appears in RD_T^P. $\Pr(D_T) = |D_T|/|RD_T^P|$, where $|D_T|$ is the total frequencies of D_T and $|RD_T^P|$ is the total frequencies of all destinations. If T is a vacant taxicab, then operations return 0 as the Distance Dispersion, given that no distance exists for a vacant taxicab.

6. Evaluation

The sample dataset, which contains 4.5 million GPS raw records of 14747 taxicabs, is used to test our recommendation system. Owing to the large size of the dataset, we find a major amount of errant records. Two main errors exist: (i) abnormal error (e.g., although the state value is 1, which means the taxicab is moving, the continuous GPS records show that the latitude and longitude are maintained, which is illogical) and (ii) matching error (after matching with the electronic map, the GPS coordinates indicate that a taxicab is off the road) [39].

These errors may result from different causes, such as GPS device malfunctions, software issues, and human factors. Before data processing, we clean the original data using simple preprocessing operations to delete abnormal and invalid GPS records.

6.1. Evaluation Setup. In this study, *VOT* compares three clustering algorithms (K-means, DBSCAN, and BIRCH) in basic and advanced models. The taxi-manned trajectory distributions, which show real passenger requests, can be obtained based on the historical GPS datasets. Real requests, which occurred in the dataset at one day, are regarded as future requests to test our recommendation system. Based on a specific manned trajectory, for example, take-in time T_x, origin area O_x, drop-off time T_y, and destination area D_y, in the taxi-manned trajectory distributions, a passenger request (request time T_x, origin O_x, and destination D_y) can be generated.

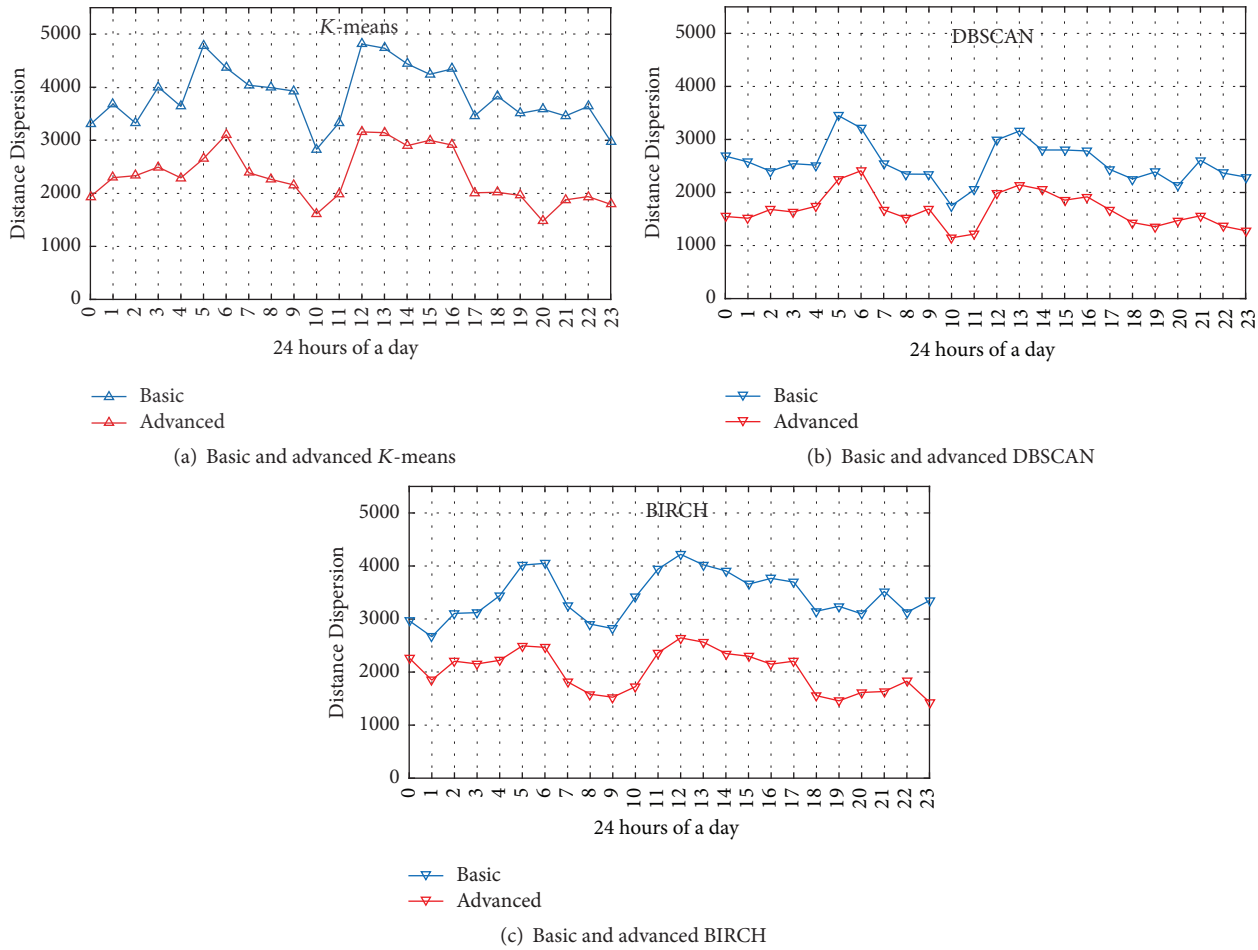
All recommendation algorithms match this actual request with the real-time GPS records for a nearby taxicab set T based on the trajectories of taxicabs in the dataset for a particular day. If vacant taxicabs exist in T, all recommendation algorithms suggest the closest vacant taxicab to passengers. Otherwise, basic K-means calculates the Distance Dispersion for every occupied taxicab in T based on the basic model and the K-means algorithm in Section 4.2 and then recommends the occupied taxicab with the minimum attribute value. Other algorithms function similarly, except that these algorithms calculate Distance Dispersion based on different clustering algorithms (DBSCAN and BIRCH) and different models (advanced models).

Distance Dispersion is regarded as a key metric to show the efficiency of taxicab service, which is obtained by $(\mathrm{EM}_{D_T}^{D_P} + \mathrm{MH}_{D_T}^{D_P})/2$; this metric is used to evaluate the closeness between passenger and taxicab destinations. For vacant taxicabs, the Distance Dispersion is 0; for occupied taxicabs, we compare and recommend the occupied taxicab with the minimum Distance Dispersion to passengers. Hence, Distance Dispersion can provide a recommendation which maximizes passengers' interests for both carpooling and conventional taxicab services.

What is more, we justify carpooling services by showing reduced total mileage (%). Unlike Distance Dispersion which concentrates on the interests of an individual passenger, reduced total mileage is used to calculate how much total mileage can be reduced (leading to less gas exhaust emissions and less traffic congestion) by an efficient system recommending more suitable taxicabs in Occupied Status for passengers. Supposing M is the total mileage for individually delivering all passengers and m is the total mileage for delivering all passengers with either conventional taxi or carpool service, then the percentage of reduced mileage equals $(M - m)/M$.

In order to prove the superiority of Distance Dispersion, we use actual detour ratio to evaluate *VOT*, which is regarded as a key metric to show the efficiency in other recommendation systems. Compared to conventional taxi service, carpooling service has a detour distance (ActualDistance − DirectDistance). Thus, actual detour ratio can be obtained by (ActualDistance − DirectDistance)/DirectDistance.

Then, we propose a new parameter, called Real Prophecy Distance (RPD), to demonstrate the ratio of correctly predicted destinations, which is obtained by quantifying this distance between true destinations and forecasted cluster centers.



(a) Basic and advanced K-means

(b) Basic and advanced DBSCAN

(c) Basic and advanced BIRCH

Figure 6: Distance Dispersion (M).

We evaluate *VOT* at different cluster numbers and various region sizes, according to the above metrics. This evaluation leads to different service effects in terms of the same algorithm. The default setting of cluster number is 5, and the default setting of region length is 600 M. For the entire dataset, we use the real requests from a one-day dataset and test all the algorithms with the trajectories of taxicabs on other days. The average results are reported.

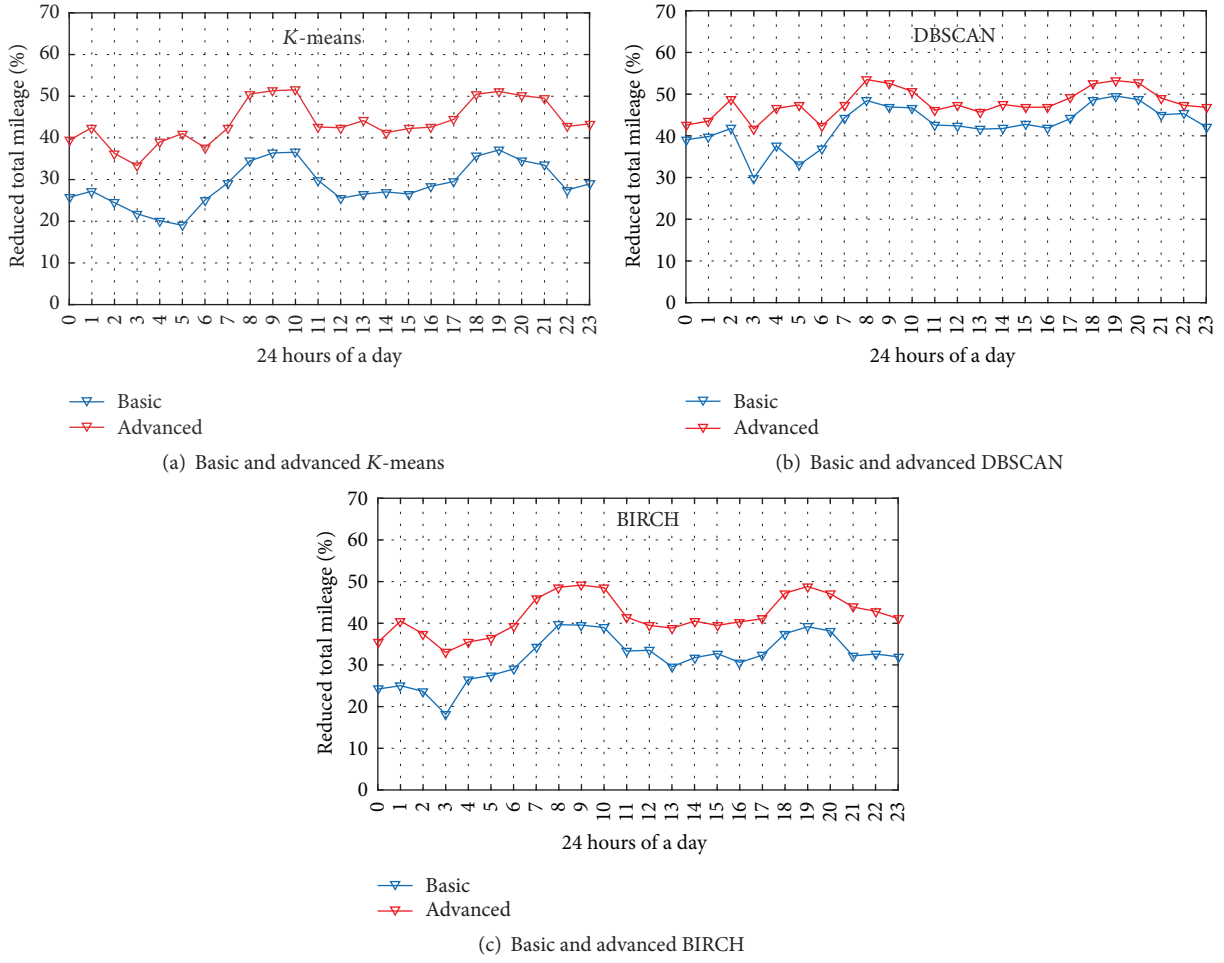
6.2. Distance Dispersion. In this subsection, we investigate the average Distance Dispersion performance.

Figure 6 shows the average Distance Dispersion in different 1 h time slots of one day. During rush hours, such as 8:00 to 10:00 AM and 18:00 to 20:00 PM, the average Distance Dispersion for all versions is lower than during nonrush hours, such as 1:00 to 7:00 AM. This result is due to the fact that passengers during rush hours have more fixed destinations and that more historical GPS data are available for predictions. Therefore, our recommendation system can more accurately predict the destinations of occupied taxicabs by context information and manned trajectory distributions.

A comparison of the three clustering algorithms indicates that DBSCAN has the best performance, with a minimum Distance Dispersion, and performs well in both basic (2.560 km) and advanced (1.671 km) scenarios, which effectively guarantees the interests of passengers. That is because DBSCAN can find clusters of arbitrary shapes, which provides it with the highest prediction accuracy. K-means has a good carpool quality in the advanced model, but the performance is poor in the basic model, with a large difference at 1.524 km. That is because many abnormal and worthless data seriously interfere with K-means in the basic model.

6.3. Reduced Total Mileage. In this subsection, we evaluate the performance of *VOT* through the percentage of reduced total mileage (%).

Figure 7 shows the percentage of reduced total mileage in different 1 h time slots. During rush hours, such as 8:00 to 10:00 AM and 18:00 to 20:00 PM, the percentages of reduced total mileage for all six schemes are higher than those on nonrush hours, especially 1:00 to 7:00 AM. This result is attributed to the increased carpooling service demands during rush hours compared to those on nonrush hours. Meanwhile, with more accurate carpooling recommendations for passengers, our recommendation system also leads to a much bigger



(a) Basic and advanced K-means

(b) Basic and advanced DBSCAN

(c) Basic and advanced BIRCH

FIGURE 7: Reduced total mileage (%).

reduction in the total mileage to deliver the same number of passengers than the reduced total mileage on nonrush hours.

In both K-means and BIRCH algorithms, the advanced model outperforms the basic one by 15.06% and 10.05% on average, respectively, indicating the superiority of the advanced model. With high carpool quality, DBSCAN is not sensitive to basic and advanced scenarios, which confirms our previous observations. From the overall view, DBSCAN is the best choice because of its stable and high carpool quality with 47.84% in reduced total mileage on average in the advanced model. Nevertheless, K-means outperforms DBSCAN at some hours in the advanced model, such as 9:00-10:00 and 21:00.

6.4. Actual Detour Ratio. Figure 8 shows the performance for the average actual detour ratio in different 1 h time slots of one day. During the busy commuting time, such as 8:00 to 10:00 AM and 18:00 to 20:00 PM, the average actual detour ratio for all three algorithms in the advanced model is higher than those on nonbusy hours, such as 1:00 to 7:00 AM. The variation trend of *VOT* is almost the same as that of similar researches.

With the best performance among the three algorithms, the actual detour ratio (%) of DBSCAN at any 1 h time slots of one day is no more than 10%, which is clearly superior to other researches in the busy commuting time. Then, although K-means and BIRCH do not have a good performance, their worst cases are still no more than 15%,

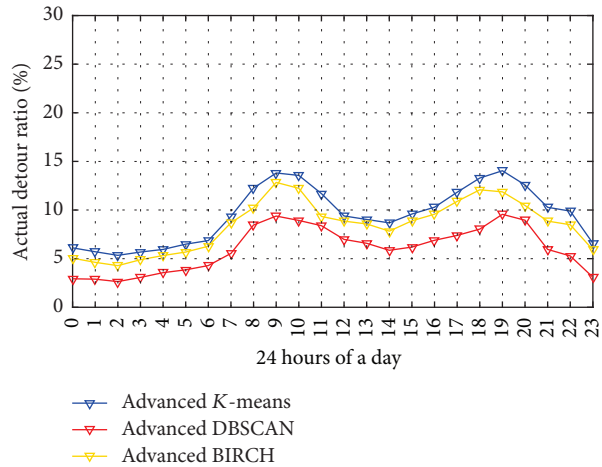
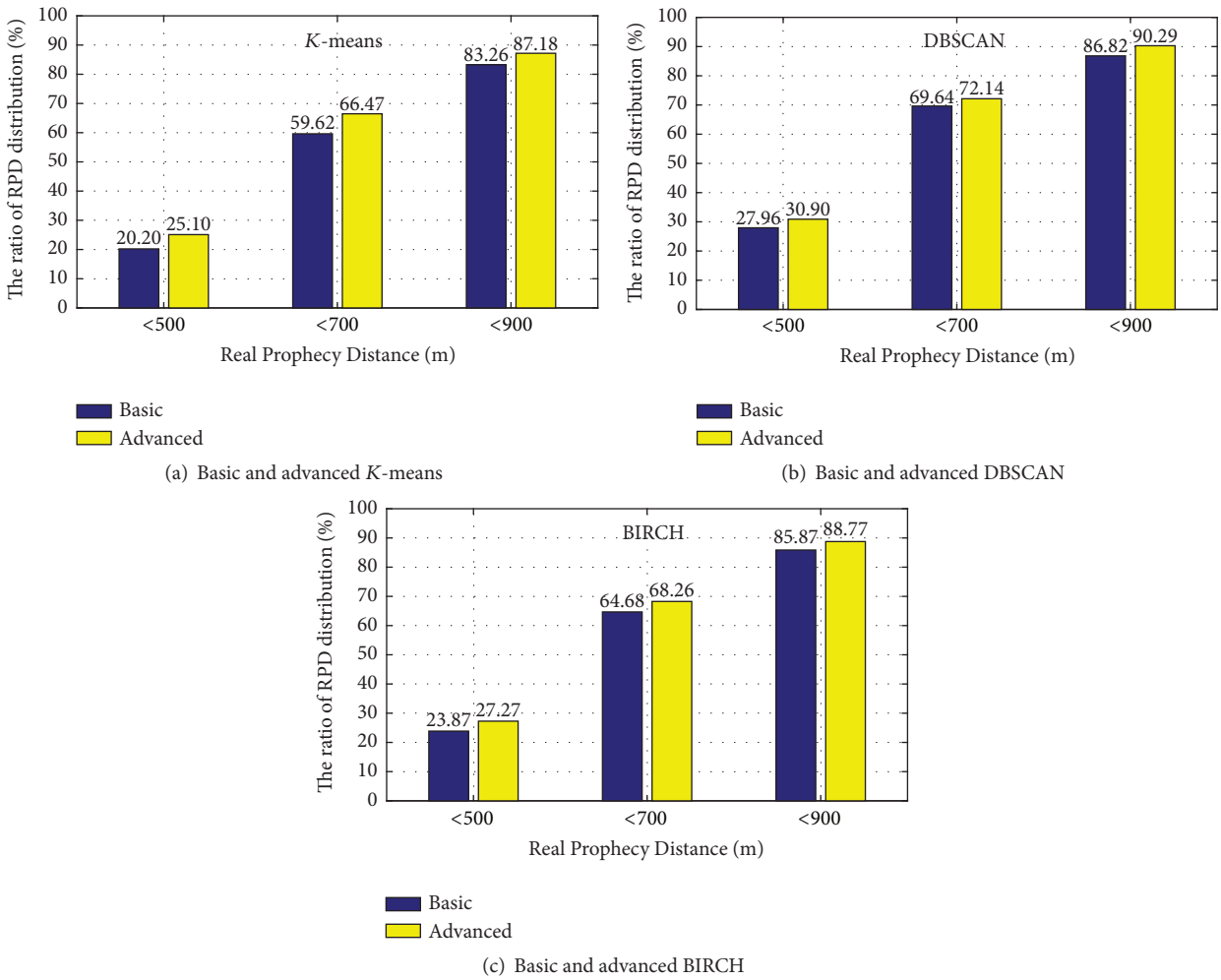


FIGURE 8: Actual detour ratio (%).



(a) Basic and advanced *K*-means

(b) Basic and advanced DBSCAN

(c) Basic and advanced BIRCH

Figure 9: Real Prophecy Distance (M).

14.04%, and 12.84% respectively. What is more, there is only 3.48% difference on average between advanced DBSCAN with the best performance and advanced *K*-means with the worst performance. In other words, all versions of *VOT* in the advanced model can fully guarantee and control the actual detour ratio.

Based on the results of the above supplementary experiments, we demonstrate that *VOT* can perform well in both the interests of passengers (actual detour ratio (%)) and the mitigation of gas exhaust emissions (reduced total mileage (%)). Therefore, Distance Dispersion is regarded as a key metric to show the efficiency of conventional and carpooling service in *VOT*, instead of actual detour ratio (%).

6.5. Real Prophecy Distance Distribution. Figure 9 shows the percentage of Real Prophecy Distance distribution under the default region length (600 M).

The distributions (<900 M) of RPD for six versions are all over 85% (except for basic *K*-means, 83.26%), especially advanced DBSCAN with 90.29%. Remarkably, because the default region length is set to 600 M, the worst condition of the RPD distributions (<900 M) is that there is only less than two regions between true destinations and forecasted cluster centers. What is more, the distributions (<500 M) of RPD for six versions are almost all over 25%, in which advanced DBSCAN has the best performance with 30.90%. Notably, RPD, which is less than 500 M, means only one situation: the predicted cluster centers are in the same region as the real destinations (or adjacent when region length is 400 M). In other words, the prediction result must be absolutely correct, if RPD is less than 500 M.

For *K*-means, DBSCAN, and BIRCH, the advanced model outperforms the basic model by 5.22%, 2.97%, and 3.29% on average, clearly indicating the superiority of the advanced model. A comparison of these clustering algorithms suggests that DBSCAN has the best performance. And DBSCAN works well in the three different RPD distributions (<500 M (30.90%), <700 M (72.14%), and <900 M (90.29%)), which clearly demonstrates the prediction accuracy of *VOT*. These results confirm that *VOT* is actually able to guarantee high prediction accuracy.

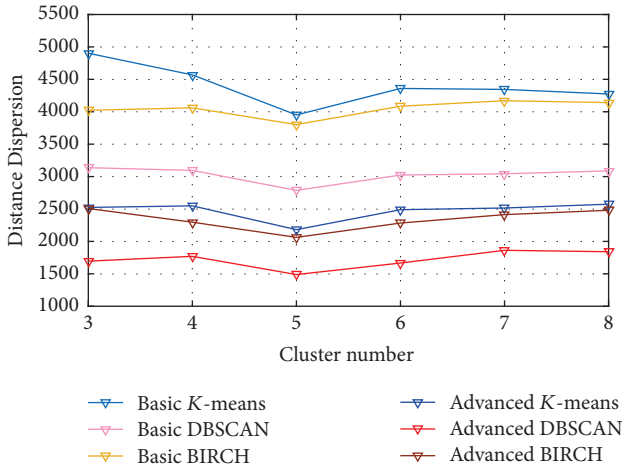


Figure 10: Distance Dispersion versus cluster number.

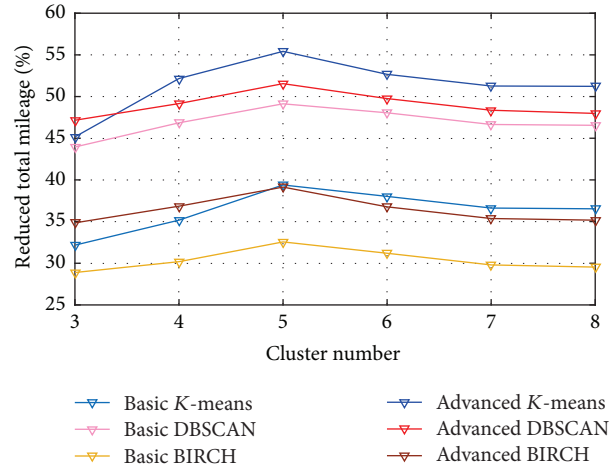


Figure 11: Reduced total mileage (%) versus cluster number.

6.6. Cluster Number Effect. In this subsection, we learn the influence of recommendation radius on *VOT* performance at 9:00 AM of one day.

6.6.1. Distance Dispersion with Different Cluster Numbers. Figure 10 shows the effect of different cluster numbers on the performance of the 6 schemes in terms of Distance Dispersion. We change the cluster number from 3 to 8, which in turn alters the number of destinations to be used to summarize the distribution characteristics of occupied taxicabs.

For all six visions of *VOT*, the Distance Dispersion under the advanced model is invariably better than that in the basic model. That is because better recommendations are provided to passengers by eliminating the worthless candidate destinations in the former. Minimum Distance Dispersion is achieved when the cluster number is 5, and the increase for 6 versions of *VOT* slows down when the cluster number is close to 8. Compared with the numbers 3 and 8 that cannot precisely generalize the characteristics of destination distribution, the number 5 is consistent with the destination distribution of a vast majority of taxicabs.

6.6.2. Reduced Total Mileage (%) with Different Cluster Numbers. Figure 11 shows the effects of different cluster numbers on the percentage of reduced total mileage at 9:00 AM of one day.

The maximum reduced total mileage occurs when the number of the clusters is 5. When the cluster number is close to 8, the decrease for 6 versions of *VOT* slows down. In other words, the minimum Distance Dispersion and the maximum reduced total mileage, which indicate the best carpool quality, occur at 5 at the same time. Thus, we recommend that the number of clusters be set to 5 for enhanced carpool quality. And in the advanced model, K-means outperforms DBSCAN in terms of reduced total mileage at 9:00 of one day, which confirms our previous observations.

6.7. Region Length Effect. In this subsection, we study the effect of recommendation radius on *VOT* performance for 24 h on one day in the advanced model. Due to the great similarity in tendency of the three algorithms, we just present the performance in K-means algorithm.

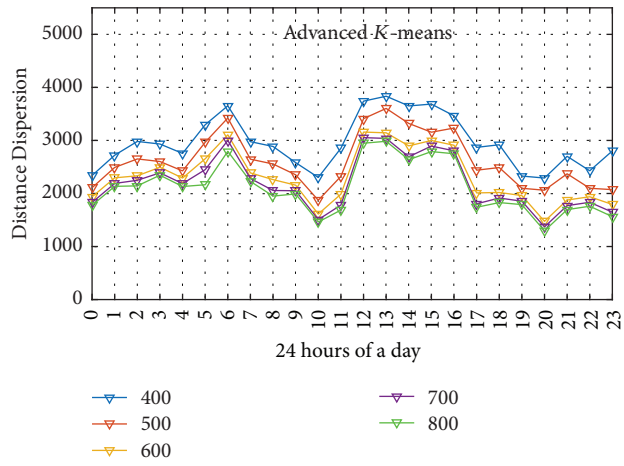


Figure 12: Distance Dispersion in advanced K-means versus region length.

6.7.1. Distance Dispersion with Different Region Lengths. Figure 12 shows the effect of different region lengths in advanced K-means on Distance Dispersion. We change the region length from 400 M to 800 M, which increases the size of potential taxicabs that can be recommended and the number of similar manned trajectories that can be analyzed.

For K-means, with the increase in the radius from 400 M to 800 M, the performance of *VOT* decreases. Nonetheless, the decrease slows down when the region length is close to 800 M, which is due to the fact that the radius is large enough to have a sufficient number of similar taxicab-manned trajectories and taxicabs for analysis and inference, and an even larger radius would not help. DBSCAN and BIRCH also have the same trend. But there are still different trends for K-means, DBSCAN, and BIRCH between 400 M

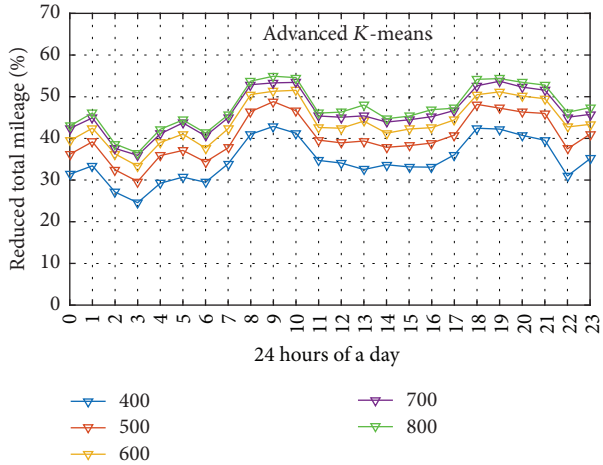


Figure 13: Reduced total mileage (%) in advanced K-means versus region length.

and 800 M, that is, 850.0760 M, 491.1766 M, and 671.8267 M, respectively.

Similar trends are maintained when the radius increases from 400 M to 800 M, such as a better performance from 18:00 to 20:00 and a worse performance from 1:00 to 7:00 AM, which verify the previous inference in the previous sections.

6.7.2. Reduced Total Mileage with Different Region Lengths. Figure 13 shows the effects of different region lengths on the percentage of reduced total mileage for 24 h on one day.

With the increase in the radius from 400 M to 800 M, the reduced total mileage of K-means in the advanced model increases given the increased carpooling service demands and the more accurate inference available. However, the increase for K-means slows down when the region length is close to 800 M. Hence, the default region length is set to 600 M because the radius is sufficiently large to provide accurate inference and calculation (only 2.76% between 600 M and 800 M), and an even larger radius is not unnecessary. DBSCAN and BIRCH also have the same trends.

The increase in region length from 400 M to 800 M leads to the largest difference in the performance of K-means between 400 M and 800 M, that is, 12.53%. By contrast, the difference in the DBSCAN performance is insignificant (i.e., 5.80%) because K-means (based on distance) is sensitive to the change in region length, whereas DBSCAN (based on density) is unresponsive to this change. Compared with K-means, the performance of BIRCH has only 9.17% increasement when the region length varies from 400 M to 800 M.

6.7.3. Real Prophecy Distance Distribution with Different Region Lengths. In this section, we evaluate the influence of region length on Real Prophecy Distance distribution under the advanced model.

Tables 2, 3, and 4 show the effect of different region lengths on the three different RPD distributions (<500 M, <700 M, and <900 M) in the advanced model.

For the three different clustering algorithms, with the increase in the region length from 400 M to 600 M, the ratio

Table 2: Real Prophecy Distance <500 M versus region length.

<500	400	500	600	Max. difference
K-means	24.0256	24.6381	25.0975	1.0719
DBSCAN	30.6759	30.7728	30.8996	0.2237
BIRCH	26.2619	26.8249	27.2711	1.0092

of RPD shows an increasing tendency. That is because a larger region length enlarges the range of a single grid, which increases the possibility that the inferred cluster centers contain the GPS records of the real destinations. But for the three different RPD distributions (<500 M, <700 M, and <900 M), the performance under 600 M outperforms that under 400 M by only 0.768%, 3.334%, and 4.740% on average. Specifically, the minimal variation tendency of the RPD distributions (<500 M) is 1.07%, 0.22%, and 1.01% for K-means, DBSCAN, and BIRCH, respectively. In other words, even if the region length is set to 400 M, all versions of *VOT* in the advanced model also can guarantee good prediction accuracy for the three RPD distributions.

In addition, in contrast to our previous comparison experiments from 400 M to 800 M in the initial manuscript, we do not carry out experiments with region lengths of 700 M and 800 M. This is because if the region length is too long, the situation satisfying the RPD distribution tends to be homogeneous. For example, when the region length is set to 800 M, the distributions (<500 M and <700 M) of RPD are quite consistent. This results in an obscure tendency of RPD distribution. Therefore, these inconclusive experiments are not executed in this section.

7. Discussion

Although *VOT* provides good carpooling performance, there is room for further enhancements. Discussed below is the system feasibility or implementability that warrants further investigation.

7.1. Changes in Existing Taxicab System. Although there is no need to build a completely new taxicab network, further optimization and promotion are necessary to the existing taxicab system for a better service. For example, a convenient two-way communication needs to be deployed between the taxicabs and the backend server, instead of one-way communication via GPS. With the development and popularization of the fourth-generation mobile communication technology, the convenience and practicability of mobile devices provide an opportunity for realization of two-way communication. Thus, we will study this respect in the further work.

7.2. An Acceptance by Passengers of Sharing the Taxi. In *VOT*, we can only realize whether the taxicab has passengers via the Status Bit of the GPS records. But if the two-way communication between the taxicabs and the backend server is realized successfully, the number of existing passengers in real-time taxicabs can be obtained by uploading the passengers' information.

Table 3: Real Prophecy Distance <700 M versus region length.

<700	400	500	600	Max. difference
K-means	61.4596	64.0293	66.4653	5.0057
DBSCAN	71.0951	71.5964	72.1402	1.0451
BIRCH	64.3057	66.4358	68.2569	3.9512

Table 4: Real Prophecy Distance <900 M versus region length.

<900	400	500	600	Max. difference
K-means	80.0751	83.6194	87.1817	7.1066
DBSCAN	89.1928	89.7317	90.2919	1.0991
BIRCH	82.7529	85.5416	88.7672	6.0143

Then, *VOT* can provide personalized carpooling options according to the preferences of passengers. For example, the acceptable number of taxi-sharing passengers is two and female only; two and male only; two and no request for male or female preference; three and female only; three and male only; three and no request for male or female preference; no request. We believe that a variety of carpool preferences options can provide passengers with more comfortable carpooling services.

7.3. The Support from Relevant Law. Through the careful and extensive investigation, currently in China, voluntary carpooling is legally a contractual relationship that belongs to the agreement of the parties' autonomy. The drivers have the obligations for ensuring the passenger safety. If man-made accidents or unforeseen events happen, the accidents should be dealt with based on the "General Principles of Civil Law" [40], "Law of Tort Liability" [41], and "Road Traffic Safety Law" [42].

There are currently no specific laws and regulations to restrict taxicab carpooling services. With the popularity of the concept of vehicle sharing, the government and a large number of researchers are actively promoting the introduction of relevant laws.

7.4. The Extra Benefit in Fleet Managers. Because reduced total mileage (%) can reach 47.84%, the cost for delivering all passengers could be significantly reduced. Namely, taxis can accomplish more delivery tasks at the same fuel costs. This could increase the income of the company and the drivers. And there are some researches [43–46] about the benefit for passengers. In the further work, the benefit for the fleet managers and passengers will be increased as an important consideration in advanced *VOT*.

8. Conclusion

In this work, we analyze, design, and evaluate a recommendation system for both carpooling and regular taxi services based on large-scale historical GPS records. Our recommendation system mines taxi-manned trajectory distributions from a historical GPS dataset. Real requests are extracted from taxi-manned trajectory distributions, and either a taxi in Wander Status with no Distance Dispersion or an occupied taxi with minimal Distance Dispersion is recommended to particular passengers. We employ a generic big-data-processing model, Spark, to efficiently handle the raw GPS dataset. Using the real-world dataset containing 14747 taxi GPS records to evaluate the system, the ratio of range (between forecasted and actual destinations) of less than 900 M can reach 90.29%, which effectively guarantees the interests of passengers. Our recommendation system can significantly reduce the total mileage (47.84% on average). Nearly half of the total mileage of the taxi is reduced, thereby effectively reducing the air and soil pollution. Meanwhile, the average reduced total mileage of 0:00 to 7:00 is increased to 45.03%, which outperforms other systems by 35%. For actual detour ratio, *VOT* and others have similar performances, even better in rush hours.

Conflicts of Interest

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

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities under Grant no. 2017QNA20.

References

[1] S. Burgaz, G. Cakmak Demircigil, B. Karahalil, and A. E. Karakaya, "Chromosomal damage in peripheral blood lymphocytes of traffic policemen and taxi drivers exposed to urban air pollution," *Chemosphere*, vol. 47, no. 1, pp. 57–64, 2002.

[2] C. M. Lytle, B. N. Smith, and C. Z. McKinnon, "Manganese accumulation along Utah roadways: a possible indication of motor vehicle exhaust pollution," *Science of the Total Environment*, vol. 162, no. 2-3, pp. 105–109, 1995.

[3] G. A. Rhys-Tyler, W. Legassick, and M. C. Bell, "The significance of vehicle emissions standards for levels of exhaust pollution from light vehicles in an urban area," *Atmospheric Environment*, vol. 45, no. 19, pp. 3286–3293, 2011.

[4] J.-C. Weng, Y.-Q. Zhai, X.-J. Zhao, and J. Rong, "Floating car data based taxi operation characteristics analysis in beijing," in *Proceedings of the WRI World Congress on Computer Science and Information Engineering (CSIE '09)*, pp. 508–512, April 2009.

[5] J. Hao, J. Hu, and L. Fu, "Controlling vehicular emissions in Beijing during the last decade," *Transportation Research Part A: Policy and Practice*, vol. 40, no. 8, pp. 639–651, 2006.

[6] M. Qu, H. Zhu, J. Liu, G. Liu, and H. Xiong, "A cost-effective recommender system for taxi drivers," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*, pp. 45–54, August 2014.

[7] L. Tang, X. Chang, and Q. Li, "The knowledge modeling and route planning based on taxi' experience," *Acta Geodaetica et Cartographica Sinica*, vol. 39, no. 4, pp. 404–409, 2010.

[8] R. Wolfler Calvo, F. de Luigi, P. Haastrup, and V. Maniezzo, "A distributed geographic information system for the daily car pooling problem," *Computers and Operations Research*, vol. 31, no. 13, pp. 2263–2278, 2004.

[9] C.-C. Tao, "Dynamic taxi-sharing service using intelligent transportation system technologies," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '07)*, pp. 3204–3207, September 2007.

[10] R. Baldacci, V. Maniezzo, and A. Mingozzi, "An exact method for the car pooling problem based on Lagrangean column generation," *Operations Research*, vol. 52, no. 3, pp. 422–439, 2004.

[11] Y. Huang, F. Bastani, R. Jin, and X. S. Wang, "Large scale real-time ridesharing with service guarantee on road networks," in *Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB '06)*, pp. 2017–2028, September 2006.

[12] S. Ma, Y. Zheng, and O. Wolfson, "T-share: a large-scale dynamic taxi ridesharing service," in *Proceedings of the 29th IEEE International Conference on Data Engineering (ICDE '13)*, pp. 410–421, IEEE, Brisbane, Australia, April 2013.

[13] Y. Fu, Y. Fang, C. Jiang, and J. Cheng, "Dynamic ride sharing community service on traffic information grid," in *Proceedings of the International Conference on Intelligent Computation Technology and Automation (ICICTA '08)*, pp. 348–352, October 2008.

[14] A. Attanasio, J.-F. Cordeau, G. Ghiani, and G. Laporte, "Parallel Tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem," *Parallel Computing*, vol. 30, no. 3, pp. 377–387, 2004.

[15] P. Healy and R. Moll, "A new extension of local search applied to the Dial-A-Ride Problem," *European Journal of Operational Research*, vol. 83, no. 1, pp. 83–104, 1995.

[16] J.-F. Cordeau, "A branch-and-cut algorithm for the dial-a-ride problem," *Operations Research*, vol. 54, no. 3, pp. 573–586, 2006.

[17] L. M. Hvattum, A. Løkketangen, and G. Laporte, "A branch-and-regret heuristic for stochastic and dynamic vehicle routing problems," *Networks*, vol. 49, no. 4, pp. 330–340, 2007.

[18] R. Bajaj, S. Ranaweera, and D. Agrawal, "GPS: location-tracking technology," *Computer*, vol. 35, no. 3, pp. 92–94.

[19] D. Zhang, T. He, Y. Liu, and J. A. Stankovic, "CallCab: A unified recommendation system for carpooling and regular taxicab services," in *Proceedings of the IEEE International Conference on Big Data (Big Data '13)*, pp. 439–447, October 2013.

[20] D. Zhang, T. He, Y. Liu, S. Lin, and J. A. Stankovic, "A carpooling recommendation system for taxicab services," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 3, pp. 254–266, 2014.

[21] H. Tian, "Data Description for UrbanCPS [EB/OL]," http://www-users.cs.umn.edu/~tianhe/BIGDATA/.

[22] Z. Gui, Y. Xiang, and Y. Li, "Parallel discovering of city hot spot based on taxi trajectories," *Huazhong Keji Daxue Xuebao (Ziran Kexue Ban)/Journal of Huazhong University of Science and Technology (Natural Science Edition)*, vol. 40, no. 1, pp. 187–190, 2012.

[23] X. Li, G. Pan, Z. Wu et al., "Prediction of urban human mobility using large-scale taxi traces and its applications," *Frontiers of Computer Science in China*, vol. 6, no. 1, pp. 111–121, 2012.

[24] S. Liu, Y. Liu, L. M. Ni, J. Fan, and M. Li, "Towards mobility-based clustering," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 919–927, July 2010.

[25] Q. Niu, T. Huan, and P. Chen, "NMCT: a novel Monte Carlo-based tracking algorithm using potential proximity information," *International Journal of Distributed Sensor Networks*, vol. 2016, Article ID 7061486, 10 pages, 2016.

[26] D. Zhang, T. He, S. Lin, S. Munir, and J. A. Stankovic, "Online Cruising Mile Reduction in Large-Scale Taxicab Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 11, pp. 3122–3135, 2015.

[27] D. Agrawal, P. Bernstein, E. Bertino et al., "Challenges and opportunities with big data," 2012, http://www.cra.org/ccc/files/docs/init/bigdatawhitepaper.pdf.

[28] W. Zhao, H. Ma, and Q. He, "Parallel K-means clustering based on mapreduce," in *Cloud Computing*, vol. 5931 of *Lecture Notes in Computer Science*, pp. 674–679, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[29] S. Gopalani and R. Arora, "Comparing Apache Spark and Map Reduce with Performance Analysis using K-Means," *International Journal of Computer Applications*, vol. 113, no. 1, pp. 8–11, 2015.

[30] D. Han, A. Agrawal, W.-K. Liao, and A. Choudhary, "A novel scalable DBSCAN algorithm with spark," in *Proceedings of the 30th IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW '16)*, pp. 1393–1402, May 2016.

[31] B.-R. Dai and I.-C. Lin, "Efficient map/reduce-based DBSCAN algorithm with optimized data partition," in *Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD '12)*, pp. 59–66, June 2012.

[32] T. Sun, C. Shut, F. Li, H. Yu, L. Ma, and Y. Fang, "An efficient hierarchical clustering method for large datasets with map-reduce," in *Proceedings of the International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT '09)*, pp. 494–499, December 2009.

[33] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large databases," *In Proceedings of the ACM SIGMOD International Conference on Management of Data*, vol. 25, no. 2, pp. 103–114, 1996.

[34] M. Zaharia, M. Chowdhury, and J. M. Franklin, "cluster computing with working sets," *HotCloud*, vol. 10, 10 pages, 2010.

[35] M. Zaharia, M. Chowdhury, and T. Das, "Fast and interactive analytics over Hadoop data with Spark," *USENIX Login*, vol. 37, no. 4, pp. 45–51, 2012.

[36] R. S. Xin, J. E. Gonzalez, M. J. Franklin, and I. Stoica, "GraphX: A resilient distributed graph system on spark," in *Proceedings of the 1st International Workshop on Graph Data Management Experiences and Systems (GRADES '13)*, June 2013.

[37] L. Gu and H. Li, "Memory or time: Performance evaluation for iterative operation on hadoop and spark," in *Proceedings of the Performance Computing and Communications 2013 IEEE International Conference on Embedded and Ubiquitous Computing*, pp. 721–727, 2013.

[38] M. Zaharia, M. Chowdhury, and T. Das, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster

computing," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. USENIX Association*, p. 2, 2012.

[39] J. S. Greenfeld, "Matching GPS observations to locations on a digital map," *Transportation Research Board 81st Annual Meeting*, 2002.

[40] "General principles of civil law, [EB/OL]," http://www.law-lib.com/law/law_view.asp?id=221001.

[41] "Law of tort liability, [EB/OL]," http://www.npc.gov.cn/huiyi/cwh/1112/2009-12/26/content_1533221.htm.

[42] "Road Traffic Safety Law, [EB/OL]," http://www.npc.gov.cn/npc/xinwen/2011-04/23/content_1653570.htm.

[43] J. Hirten and S. Beroldo, "Ridesharing programs cost little, do a lot," *Transportation Quarterly*, vol. 51, no. 2, pp. 9–13, 1997.

[44] M. Naor, "On fairness in the carpool problem," *Journal of Algorithms. Cognition, Informatics and Logic*, vol. 55, no. 1, pp. 93–98, 2005.

[45] R. B. Noland, W. A. Cowart, and L. M. Fulton, "Travel demand policies for saving oil during a supply emergency," *Energy Policy*, vol. 34, no. 17, pp. 2994–3005, 2006.

[46] M. Ajtai, J. Aspnes, M. Naor, Y. Rabani, L. J. Schulman, and O. Waarts, "Fairness in scheduling," *Journal of Algorithms. Cognition, Informatics and Logic*, vol. 29, no. 2, pp. 306–357, 1998.