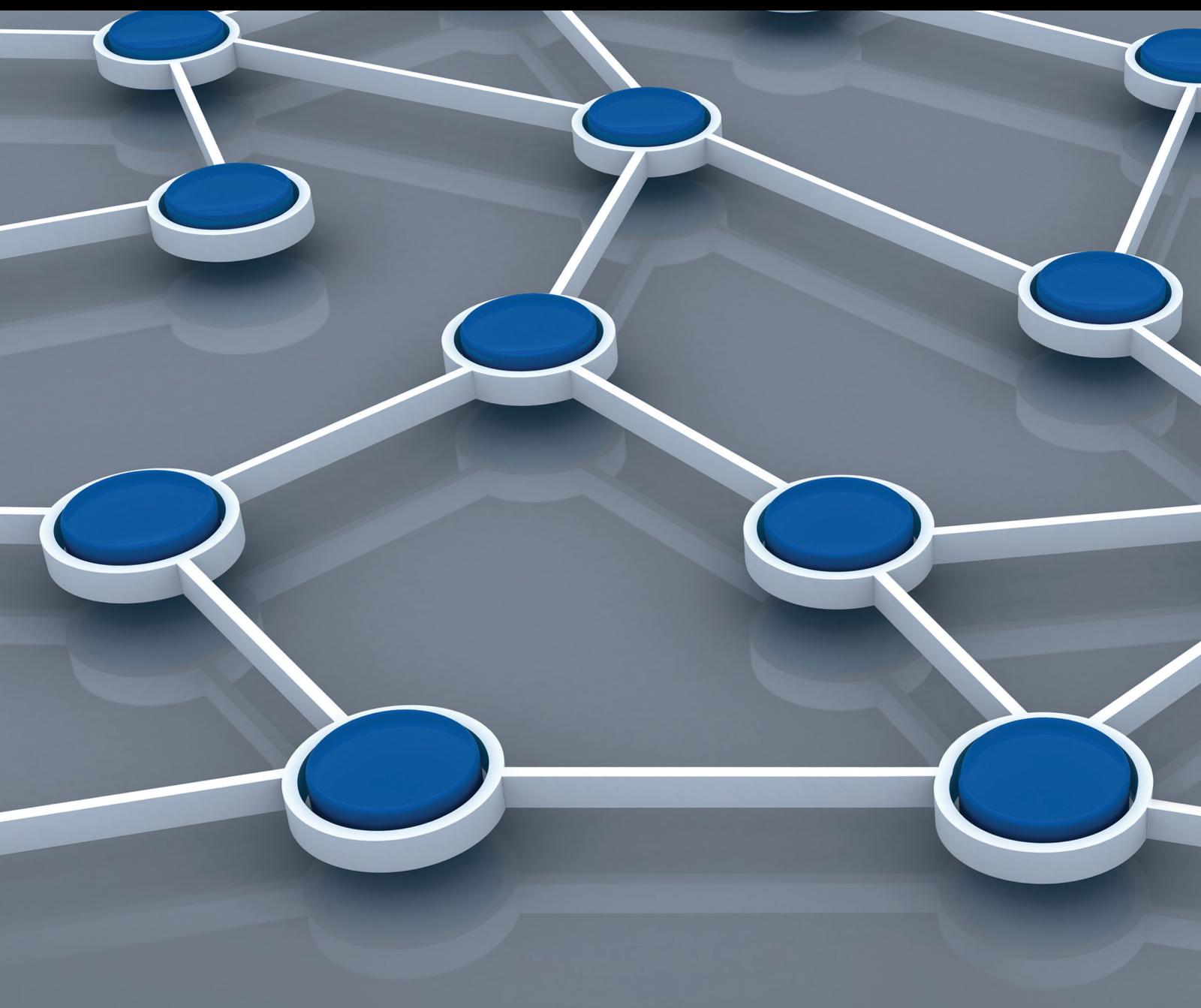


Sensing with Mobile Wireless Sensor Networks

Guest Editors: Zhongwen Guo, Minglu Li, Yu Wang, Lei Chen,
and Feng Hong



Sensing with Mobile Wireless Sensor Networks

Sensing with Mobile Wireless Sensor Networks

Guest Editors: Zhongwen Guo, Minglu Li, Yu Wang, Lei Chen, and Feng Hong



Copyright © 2014 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in "International Journal of Distributed Sensor Networks." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Editorial Board

- Jemal H. Abawajy, Australia
Miguel Acevedo, USA
Sanghyun Ahn, Korea
Mohammad Ali, USA
Jamal N. Al-Karaki, Jordan
Habib M. Ammari, USA
Javier Bajo, Spain
Prabir Barooah, USA
Alessandro Bogliolo, Italy
Richard R. Brooks, USA
James Brusey, UK
Erik Buchmann, Germany
Jian-Nong Cao, Hong Kong
João Paulo Carmo, Portugal
Jesús Carretero, Spain
Luca Catarinucci, Italy
Henry Chan, Hong Kong
Chih-Yung Chang, Taiwan
Periklis Chatzimisios, Greece
Ai Chen, China
Peng Cheng, China
Jinsung Cho, Korea
Kim-Kwang Choo, Australia
Chi-Yin Chow, Hong Kong
W.-Y. Chung, Republic of Korea
Mauro Conti, Italy
Dinesh Datla, USA
Amitava Datta, Australia
Danilo De Donno, Italy
Ilker Demirkol, Spain
Der-Jiunn Deng, Taiwan
Longjun Dong, China
Chyi-Ren Dow, Taiwan
George P. Efthymoglou, Greece
Frank Ehlers, Italy
Melike Erol-Kantarci, Canada
Giancarlo Fortino, Italy
Luca Foschini, Italy
David Galindo, France
Weihua Gao, USA
Deyun Gao, China
Athanasios Gkelias, UK
Iqbal Gondal, Australia
Jayavardhana Gubbi, Australia
Cagri Gungor, Turkey
Song Guo, Japan
Andrei Gurto, Finland
Qi Han, USA
Z. Hanzalek, Czech Republic
Tian He, USA
Junyoung Heo, Korea
Zujun Hou, Singapore
Jiun-Long Huang, Taiwan
Yung-Fa Huang, Taiwan
Xinming Huang, USA
Baoqi Huang, China
Chin-Tser Huang, USA
Wei Huangfu, China
Mohamed Ibnkahla, Canada
Tan Jindong, USA
Ibrahim Kamel, UAE
Li-Wei Kang, Taiwan
Rajgopal Kannan, USA
Sherif Khattab, Egypt
Lisimachos Kondi, Greece
Marwan Krunz, USA
Kun-Chan Lan, Taiwan
Yee W. Law, Australia
Y.-K. Lee, Republic of Korea
JongHyup Lee, Republic of Korea
K.-C. Lee, Republic of Korea
Yong Lee, USA
S. Lee, Republic of Korea
Seokcheon Lee, USA
Joo-Ho Lee, Japan
Shijian Li, China
Minglu Li, China
Shuai Li, USA
Zhen Li, China
Shancang Li, UK
Ye Li, China
Jing Liang, China
Weifa Liang, Australia
Yao Liang, USA
Wen-Hwa Liao, Taiwan
Alvin S. Lim, USA
Kai Lin, China
Zhong Liu, China
Ming Liu, China
Donggang Liu, USA
Yonghe Liu, USA
Chuan-Ming Liu, Taiwan
Zhigang Liu, China
Leonardo Lizzi, France
Giuseppe Lo Re, Italy
Seng Loke, Australia
Jonathan Loo, UK
Juan Riquelme, Spain
Pascal Lorenz, France
KingShan Lui, Hong Kong
Jun Luo, Singapore
Álvaro Marco, Spain
Jose Martinez-de Dios, Spain
Nirvana Meratnia, The Netherlands
Shabbir N. Merchant, India
Mihael Mohorcic, Slovenia
José Molina, Spain
V. Muthukumarasamy, Australia
Eduardo Freire Nakamura, Brazil
Kameswara Rao Namuduri, USA
George Nikolakopoulos, Sweden
Marimuthu Palaniswami, Australia
Ai-Chun Pang, Taiwan
Seung-Jong J. Park, USA
Soo-Hyun Park, Korea
Miguel A. Patricio, Spain
Wen-Chih Peng, Taiwan
Janez Perš, Slovenia
Dirk Pesch, Ireland
Shashi Phoha, USA
Antonio Puliafito, Italy
Hairong Qi, USA
Nageswara S. V. Rao, USA
Md. Abdur Razzaque, Bangladesh
Pedro Pereira Rodrigues, Portugal
Joel J. P. C. Rodrigues, Portugal
Jorge Sa Silva, Portugal
Mohamed Saad, UAE
Sanat Sarangi, India
Stefano Savazzi, Italy
Marco Scarpa, Italy
Arunabha Sen, USA
Xiao-Jing Shen, China
Weihua Sheng, USA
Louis Shue, Singapore

Antonino Staiano, Italy	Wen-Jong Wu, Taiwan	Ning Yu, China
Tan-Hsu Tan, Taiwan	Chase Qishi Wu, USA	Changyuan Yu, Singapore
Guozhen Tan, China	Bin Xiao, Hong Kong	Tianle Zhang, China
Shaojie Tang, USA	Qin Xin, Faroe Islands	Yanmin Zhu, China
Bulent Tavli, Turkey	Jianliang Xu, Hong Kong	T. L. Zhu, USA
Anthony Tzes, Greece	Yuan Xue, USA	Yi-hua Zhu, China
Agustinus B. Waluyo, Australia	Ting Yang, China	Qingxin Zhu, China
Yu Wang, USA	Hong-Hsu Yen, Taiwan	Li Zhuo, China
Ran Wolff, Israel	Li-Hsing Yen, Taiwan	Shihong Zou, China
Jianshe Wu, China	Seong-eun Yoo, Korea	

Contents

Sensing with Mobile Wireless Sensor Networks, Zhongwen Guo, Minglu Li, Yu Wang, Lei Chen, and Feng Hong
Volume 2014, Article ID 936830, 2 pages

An Efficient Localization Method Based on Adaptive Optimal Sensor Placement, Jin-Hee Lee, Kyeongyul Kim, Sang-Chul Lee, and Byeong-Seok Shin
Volume 2014, Article ID 983618, 11 pages

Cognitive Radio-Based Vehicular Ad Hoc and Sensor Networks, Mohammad Jalil Piran, Yongwoo Cho, Jihyeok Yun, Amjad Ali, and Doug Young Suh
Volume 2014, Article ID 154193, 11 pages

Dynamic Groups Based Adaptive DTN Routing Algorithms in Social Networks, Jixing Xu, Jianbo Li, Lei You, and Chenqu Dai
Volume 2014, Article ID 245369, 13 pages

A Flow-Partitioned Unequal Clustering Routing Algorithm for Wireless Sensor Networks, Jian Peng, Xiaohai Chen, and Tang Liu
Volume 2014, Article ID 875268, 12 pages

An Improved Group-Based Neighbor Discovery Algorithm for Mobile Sensor Networks, Qiang Niu, Weiwei Bao, and Shixiong Xia
Volume 2014, Article ID 803870, 7 pages

Mobile Beacon Based Wormhole Attackers Detection and Positioning in Wireless Sensor Networks, Honglong Chen, Wendong Chen, Zhibo Wang, Zhi Wang, and Yanjun Li
Volume 2014, Article ID 910242, 10 pages

DCEP: Data Collection Strategy with the Estimated Paths in Ocean Delay Tolerant Network, Chao Liu, Zhongwen Guo, Feng Hong, and Kaishun Wu
Volume 2014, Article ID 518439, 11 pages

Mobile Tracking Based on Support Vector Regressors Ensemble and Game Theory, Fanzi Zeng, Shaoyuan Liu, Renfa Li, and Qingguang Zeng
Volume 2014, Article ID 403927, 6 pages

Continuous Probabilistic Skyline Queries for Uncertain Moving Objects in Road Network, Shanliang Pan, Yihong Dong, Jinfeng Cao, and Ken Chen
Volume 2014, Article ID 365064, 13 pages

An Efficient Confidentiality and Integrity Preserving Aggregation Protocol in Wireless Sensor Networks, Liehuang Zhu, Zhen Yang, Jingfeng Xue, and Cong Guo
Volume 2014, Article ID 565480, 8 pages

Editorial

Sensing with Mobile Wireless Sensor Networks

Zhongwen Guo,¹ Minglu Li,² Yu Wang,³ Lei Chen,⁴ and Feng Hong¹

¹Ocean University of China, Qingdao 266100, China

²Shanghai Jiao Tong University, Shanghai, China

³University of North Carolina at Charlotte, Charlotte, NC, USA

⁴Sam Houston State University, Huntsville, TX, USA

Correspondence should be addressed to Zhongwen Guo; guozhw@ouc.edu.cn

Received 31 August 2014; Accepted 31 August 2014; Published 22 December 2014

Copyright © 2014 Zhongwen Guo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mobile wireless sensor networks (MWSNs) can be defined as a wireless sensor network (WSN) with some nodes of mobility. MWSNs are an emerging field of research in contrast to their well-established predecessor. MWSNs are much more versatile than static sensor networks as they can be deployed in any scenario and cope with rapid topology changes.

The special issue intends to give an overview of the state-of-the-art issues and solution guidelines for MWSNs. It also includes some recommended submissions on topics of MWSNs from the 7th China Conference on Wireless Sensor Networks (CWSN 2013). The special issue accepted nine high quality papers from 31 submissions. It includes the topics of network architecture, routing algorithm, localization and tracking, and data collection and query in MWSNs among the application domains of terrestrial, offshore, vehicular sensor networks and delay tolerant networks.

Four papers discuss the fundamental issues on node localization and tracking, wormhole detection, and cluster routing in MWSN. “*Mobile tracking based on support vector regressors ensemble and game theory*” written by F. Zeng et al. presents a two-step tracking strategy to mitigate the adverse effect of non-line-of-sight (NLOS) propagation for mobile node tracking. It exploits support vector regressors ensemble (SVRM) to establish the mapping of node position to radio parameters by supervising learning and models the noise as the adversary of position estimator and then estimates the mobile node position smoothing by game theory. “*An efficient localization method based on adaptive optimal sensor placement*” written by J.-H. Lee et al. proposes to track

user location by means of trilateration, using the distance between fixed nodes deployed at predetermined locations and a mobile base station with the optimal placement of the fixed nodes.

“*Mobile beacon based wormhole attackers detection and positioning in wireless sensor networks*” written by H. Chen et al. proposes to detect wormhole through the communication between the mobile beacon and each of the static beacons and located the wormhole by determining the intersection point of the chord perpendicular bisector. “*A flow-partitioned unequal clustering routing algorithm for wireless sensor networks*” written by J. Peng et al. proposes to choose cluster heads with more residual energy and larger overlapping degree and related routing procedure to achieve energy efficiency and balance.

Two papers focus on neighbor discovering and routing under neighbor social relationship. Q. Niu et al. proposes “*An improved group-based neighbor discovery algorithm for mobile sensor networks*,” including the network model and the algorithm which dynamically adjusts node active time based on spatial property to reduce discovery delay of mobile sensor networks. “*Dynamic groups based adaptive DTN routing algorithms in social networks*” written by J. Xu et al. proposes an adaptive routing algorithm, taking full use of gregariousness characteristics of moving nodes. It improves the routing efficiency by dynamically dividing nodes into different social groups according to social relationship and proposes flooding and redundancy control model to select fewer but better relay nodes to reduce message redundancy.

Two papers concern the architecture and query issues for vehicular ad hoc and sensor networks. “*Cognitive radio-based vehicular ad hoc and sensor networks*” written by M. J. Piran et al. proposes the networking paradigm for vehicular communication by utilizing wireless sensor nodes based on cognitive radio technology. “*Continuous probabilistic skyline queries for uncertain moving objects in road network*” written by S. Pan et al. studies and presents the continuous probabilistic skyline query algorithm for uncertain moving objects in road network.

The article “*DCEP: data collection strategy with the estimated paths in ocean delay tolerant network*” written by C. Liu et al. provides the network architecture of hybrid networks for offshore surveillance, exploiting underwater acoustic sensor networks to sample the offshore environment and delay tolerant networks of ships to collect sensor data. It also includes the optimized data distribution and routing strategy to improve the data collection rate.

Acknowledgment

We would like to thank our colleagues for their help and assistance in providing extraordinary high quality peer reviews for this special issue.

*Zhongwen Guo
Minglu Li
Yu Wang
Lei Chen
Feng Hong*

Research Article

An Efficient Localization Method Based on Adaptive Optimal Sensor Placement

Jin-Hee Lee, Kyeongyul Kim, Sang-Chul Lee, and Byeong-Seok Shin

Department of Computer Science and Information Engineering, Inha University, Room 1403, Hi-Tech Building, 100 Inha-ro, Nam-Gu, Incheon 402-751, Republic of Korea

Correspondence should be addressed to Byeong-Seok Shin; bsshin@inha.ac.kr

Received 27 February 2014; Revised 28 June 2014; Accepted 28 June 2014; Published 27 August 2014

Academic Editor: Feng Hong

Copyright © 2014 Jin-Hee Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a ZigBee-based localization method that estimates the distance between ZigBee nodes employing the strength of wireless signal. It enables us to track the location of a user by means of trilateration, using the distance between fixed nodes deployed at predetermined locations and a mobile base station. In addition, we propose a method to determine the optimal placement of the fixed nodes using minimum Bayes error estimation based on Gaussian distributions. As a result, this method can accurately estimate the position of the mobile base station with a minimum number of fixed nodes.

1. Introduction

In recent decades, positioning systems such as GPS, dead reckoning techniques, and intensively installed RFID tags have been used to localize a user continuously. Although GPS localization is a popular method, it is not suitable for pedestrians since the accuracy of GPS data is relatively low (more than 10 m). Therefore, we propose a more accurate localization method using the signal strength between ZigBees, instead of GPS.

We use range-based ZigBee as landmarks and apply trilateration using wireless sensor network with those nodes to estimate the position of the user. Although arranging of the numerous landmarks improves the accuracy of localization, it increases the cost. Accordingly, we propose an efficient localization method using minimum Bayes error estimation for optimal placement by generating and modelling the signal. In order to verify the availability of the proposed method, we applied it to a guidance system for visually impaired person that was developed in our previous studies [1]. As a result, the method helps pedestrians arrive at their destination safely.

We summarize related research in Section 2. In Section 3, we describe the ZigBee-based localization method and the optimal placement of sensors in detail. In Section 4, we

present the experimental results while applying the proposed method to the prototype system. Finally, we conclude our study in Section 5 and discuss future work.

2. Related Works

Localization methods can be categorized according to the underlying wireless-communication systems such as methods using Wi-Fi, ZigBee, and UWB (Ultra wideband) [2–5]. In addition, these methods are classified into several types depending on the implementation method such as Cell-ID location technique, screen analysis method, triangulation, and stochastic modelling method [6, 7].

While using Wi-Fi, it is unnecessary to install other devices and it provides position data because of the access points (APs) that have been placed for communication. However, the accuracy of the method is lower than that of other methods and it may cause more interference [8–10]. An alternative is ZigBee, which can be implemented with small-sized tags and its installation is simple. However, it requires an accurate propagation model and a detailed map [11–13]. UWB provides relatively high precision and low power consumption, but it is not suitable to be used as an outdoor landmark because it requires a large bandwidth and has a short transmission range [14, 15].

The Cell-ID location technique called the proximity method is the simplest technique for localization [16]. It determines the position of a user by checking whether that user is in a specified area called a *cell*. However, the cost of the method increases as the density of cells increases and the accuracy of localization depends strictly on the size of the cell. Triangulation is the most common method for localization by means of the angles from three positions to the user [17–19]. Its accuracy varies depending on existence of obstacles and the radio environment such as the reflection and refraction of radio wave. Another position estimation method is fingerprinting, which is based on a probabilistic model. It selects a large number of sample points and stores the signal characteristics of these points in a database. It estimates the position of user by searching for a sample point with similar characteristics to the signal of the user [20, 21]. This method shows the highest accuracy compared to other methods since it estimates position based on the location of the facilities as well as the orientation of the user. However, this method must acquire various signal characteristics for many sample points several times and update them whenever the location of obstacles changes. Further, searching the database is very complicated.

Our proposed method does not require many hardware devices and only uses range-based ZigBee with relatively high accuracy compared to universal ZigBee. In addition, we propose an optimal placement method based on Gaussian distribution that is able to cover the entire area with a minimum number of ZigBee nodes. Consequently, it provides exact position tracking with low cost. Furthermore, we verified that it helps persons with visual impairment to arrive at their destination safely outdoors by applying this method to a portable guidance system that we proposed previously [1].

3. A Localization via Efficient Placement of Wireless Sensors

In this section, we explain the trilateration technique for tracking the position of users and the optimal placement method of landmarks.

3.1. Localization Based on Wireless Sensor Network. Position estimation methods are categorized into range-based method that uses the distances of nodes and range-free method that uses inferred distances among neighbor nodes. We exploit the range-based method as it is regarded as being highly accurate when compared to other methods. We use SDS-TWR (symmetrical double-sided two way ranging) technique to measure the distance from the fixed node to the mobile (portable) base station [22]. We define ranging frame (*frame*) that means a special packet transmitted to measure the distance between two nodes. As shown in Figure 1, to measure the distance between node A and node B in the sensor network, we evaluate propagation time T_p . When the node A transfers a *frame* packet to node B, sending time stamp T_s is stored at the node A and receiving time stamp T_r is saved at the node B. If B receives the *frame* packet, it sends the

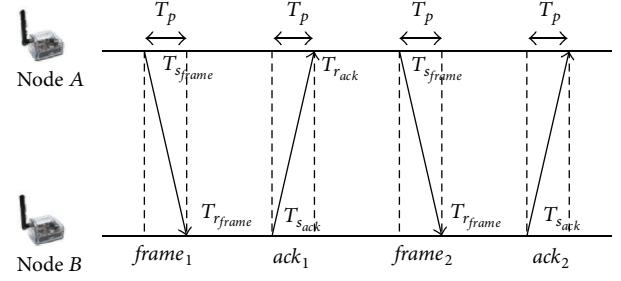


FIGURE 1: Symmetric double-sided two-way ranging (SDS-TWR) methodology.

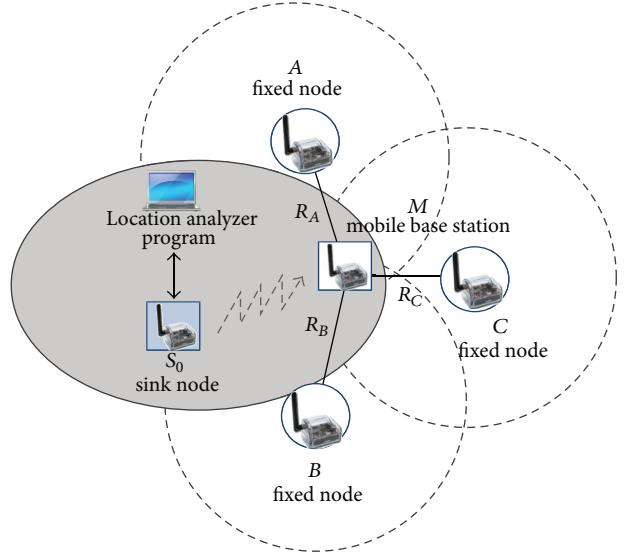


FIGURE 2: Trilateration: location tracking of mobile base station using three fixed nodes.

ack packet to node A (see (1)). It measures the distance by multiplying the propagation speed and the propagation time between two nodes. Therefore, the technique enables us to measure accurate distance because it takes advantage of the difference between a packet's transmission and reception time between two nodes.

$$T_p = \frac{\{(T_{r_{ack}} - T_{s_{frame}}) - (T_{s_{ack}} - T_{r_{frame}})\}}{2}. \quad (1)$$

We use a centralized star network topology since the fixed nodes only communicate with the mobile base station. After the closest three nodes from the mobile base station are selected, we perform trilateration to determine the relative position of the mobile base station. The position is measured by three fixed nodes and equations for the three circles derived from them [23]. As shown in Figure 2, the position of the mobile base station M is calculated from distances R_A , R_B , and R_C , and the position of three fixed nodes A, B, and C. S_0 plays a gateway role as a sink node that is connected to the location analysis program. It transmits commands and data to mobile base station. The position of the user is calculated using the localization program operated on computer with the mobile base station in real time.

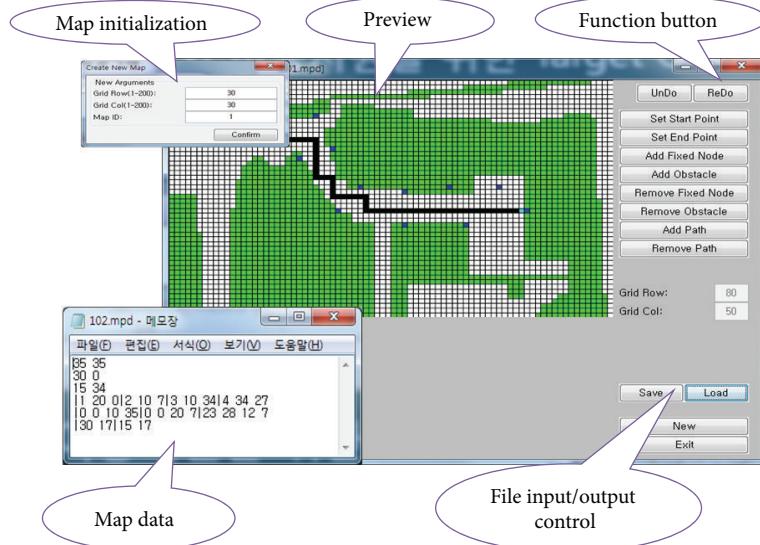


FIGURE 3: An example scene of the map authoring tool.

We use a grid-based map building method that can be implemented easily in order to estimate the location of the user [24]. The map building for visually impaired person is required for accurate modeling of hazardous objects in surrounded environment. As shown in Figure 3, we have implemented the map authoring tool. It enables us to display detail information such as position of the fixed nodes, departure point, destination, POI (point of interest), and obstacles.

Figure 4 shows the concept of the proposed localization method. The fixed nodes used as outdoor landmark are installed by applying Bayes error estimation (see Section 3.2). Map database presents the position coordinate and ID (identification) of fixed nodes. Localization of mobile base station is estimated by means of trilateration and map building method.

3.2. Optimal Placement Based on Minimum Bayes Error Estimation. In order to estimate the position of the user accurately with a minimum number of fixed nodes, we use minimum Bayes error estimation based on Bayesian decision theory with Gaussian distribution. We virtually arrange fixed nodes densely on all possible paths. The method finds the layout with maximization of cumulative minimum Bayes error (MCMB) by sequentially removing the node with a maximum error at each stage.

We generate a signal so as to reflect the characteristics that strength of ZigBee signal decreases depending on the distance. We consider simulated signals from an i th ZigBee node as a bivariate Gaussian distribution that is modelled by parameter with $N(M, \Sigma)$, as shown in Figure 5(b). The signal is circularly radiated with the given distribution as described in Figure 5(c), which can be applied to the Bayes error estimation method to measure optimal displacement.

As depicted in Figure 5(a), the signal range of a node in polar coordinates is defined as a circle with radius r as follows:

$$r = x_1 \cos \theta + x_2 \sin \theta, \quad (2)$$

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & \cos \theta \\ \sin \theta & \sin \theta \end{bmatrix} \begin{bmatrix} r_{\text{off}} \\ r_{S_0} \end{bmatrix}. \quad (3)$$

The signal consists of m concentric circles with offset r_{off} . A signal is emitted at regular intervals θ in the range from $-\pi$ to $+\pi$ in a circle. Equation (2) calculates radius r , and (3) evaluates the position of a point r_{e_m} on the circle. We define linear discrimination function $h_{qr}(X)$ using the given signal models of two ZigBees Z_q and Z_r as follows:

$$h_{qr}(X) = (M_r - M_q)^T \Sigma^{-1} X_{qi} + \frac{1}{2} (M_q^T \Sigma^{-1} M_q - M_r^T \Sigma^{-1} M_r) \geq \frac{\omega_r}{\omega_q} \ln \frac{P_q}{P_r}, \quad (4)$$

where the $X_{qi} = [x_{qi1} \ x_{qi2}]$ represents an i th signal point as random vector, which is generated by q th ZigBee node Z_q that has the prior probability $P_q = 0.5 = 1 - P_r$. The ω is class variable and X_i can be classified to class ω_q or ω_r in binary classification. The sample parameters $M_q = [\mu_{q1} \ \mu_{q2}]$ and Σ are the sample mean vector and covariance matrix, which are estimated with signals from the node Z_q . The sample parameter estimations of given distribution are defined as follows:

$$\mu_{q1} = \frac{1}{N} \sum_{i=1}^N x_{qi1}, \quad \mu_{q2} = \frac{1}{N} \sum_{i=1}^N x_{qi2}, \quad (5)$$

$$\Sigma_q = \frac{1}{N} \sum_{i=1}^N (X_{qi} - M_q)(X_{qi} - M_q)^T = \Sigma_r = \Sigma. \quad (6)$$

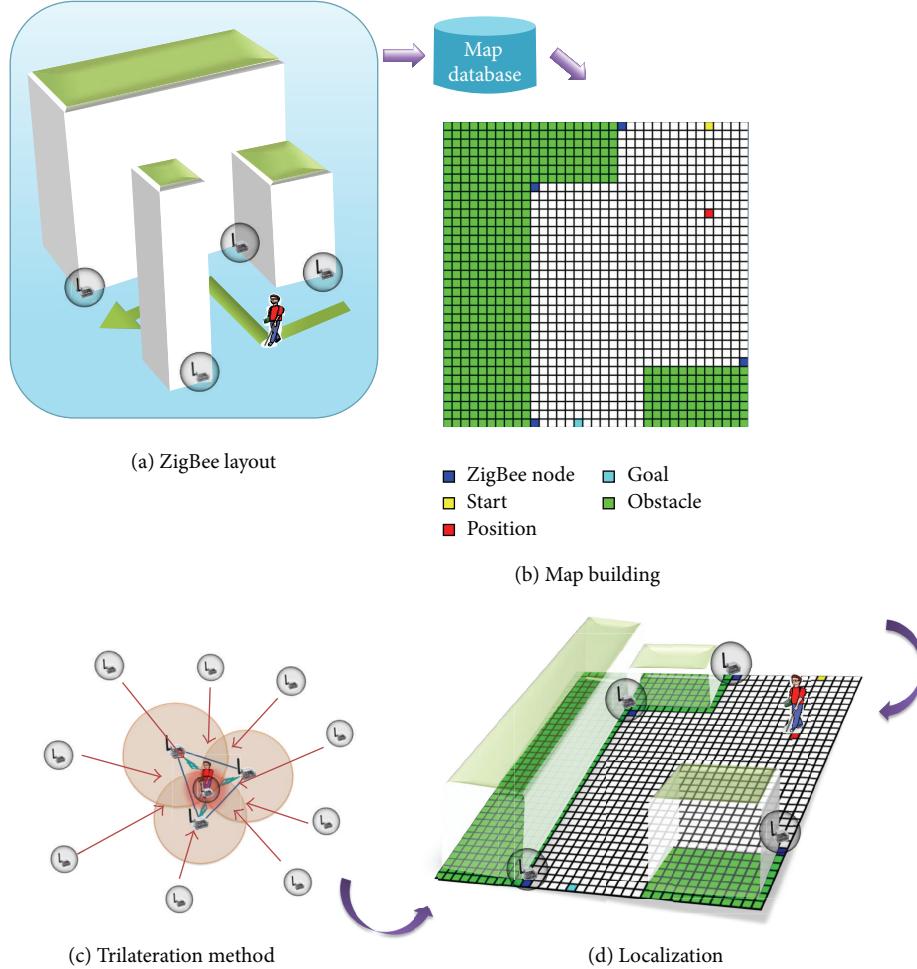


FIGURE 4: Localization concept: (a) layout of replaced ZigBees to be served as POIs, (b) a map indicating POIs and obstacles, (c) trilateration method using adjacent three ZigBees of user, and (d) localization algorithm to estimate position of user.

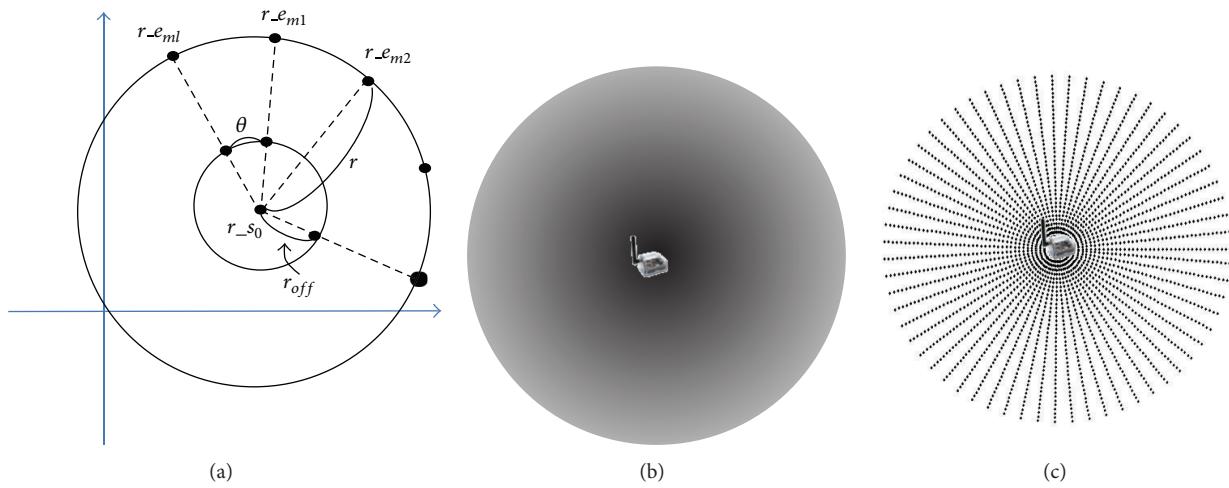


FIGURE 5: Signal generating simulation: (a) signal generation in polar coordinates, (b) signal model based on the Gaussian distribution, and (c) signal simulation for the MCMB.

In this case, the covariance matrix is denoted by Σ because we use homogeneous modules with no constraints in each node and hence the values of Σ_q and Σ_r are the same (see (6)). N is the total number of signal points. Since every node is considered as same emitting conditions (i.e., same distribution functions), (4) satisfies linear decision boundary. $\ln P_q/P_r$ is called the likelihood ratio as a threshold constant for decision [25], which is always the 0 as all the priors are set to equal probability.

Equation (4) satisfies the minimum Bayes error criterion to determine the decision boundary between the signal points from two nodes. As shown in Figure 6(a), the example case of the signal points of two nodes are represented by the classes, ω_q and ω_r , and the Bayes error regions of the two classes are denoted by ε_q and ε_r , respectively. In this figure, the decision boundary to distinguish signal points of two nodes with minimum Bayes error criterion is displayed on the dotted line on one-dimensional case. The minimum Bayes error estimation using linear decision function $h_{qr}(X)$ in two-dimensional space (Figure 6(b)) is defined as follows:

$$\begin{aligned}\varepsilon_q &= \iint_0^\infty p_h(h_{qr} | \omega_q) dh, \\ \varepsilon_r &= \iint_{-\infty}^0 p_h(h_{qr} | \omega_r) dh,\end{aligned}\quad (7)$$

where ε is the minimum Bayes error that is obtained by sum of a region for false classification to an opposite class. To find optimal placement of all the ZigBee nodes in a field, we propose the MCMB method based on adaptive local optimization using minimum Bayes error criterion in (7). The following equations describe the MCMB which adaptively detects over-emitted area (i.e., excessive overlapped area) using discretized estimation of Bayes error:

$$\begin{aligned}\psi(X_i, h_{qr}) &= \begin{cases} 1 & \text{if } h_{qr}(X_i) = r \\ 0 & \text{otherwise,} \end{cases} \\ \hat{\varepsilon}_q &= \sum_{r=1}^K \sum_{i=1}^N \psi(X_i, h_{qr}),\end{aligned}\quad (8)$$

where q and r are class variables and ψ is binary function, which returns one to false classification. That requires parameters as decision function h and feature vector X . $\hat{\varepsilon}_q$ is a cumulative minimum Bayes error which is calculated with N number of signal points from K set of nearest ZigBee nodes such as $\{Z_1, Z_2, \dots, Z_K\}$. The set of nearest ZigBee nodes is selected by whether range of signal overlaps or not with signal of current node.

The MCMB is determined by detecting node which has maximum $\hat{\varepsilon}$ within calculated cumulative error set of $\{\hat{\varepsilon}_1, \hat{\varepsilon}_2, \dots, \hat{\varepsilon}_K\}$, and then this procedure iteratively performs until $\hat{\varepsilon}$ reaches to termination threshold ρ . We set the two termination conditions for optimization process for trilateration, which are when overlapped regions are less than two nodes and when the maximum value is less than the threshold. We called a node to be removed as RN and Figure 7 shows an example of the MCMB method that adaptively

TABLE 1: Comparison of distance accuracy according to sampling rate.

Sampling rate (Hz)	Signal loss (%)	Accuracy (%)
10	1.67	73
20	2.33	80
40	2.17	92
50	9.33	84
100	36.67	56

determines an RN at each step and then removes it. After applying MCMB, only the optimal nodes for trilateration remain.

4. Experimental Results

4.1. Accuracy Estimation Using Range-Based ZigBee. We carried out experiments with two different types of ZigBee modules to verify of the effectiveness of the proposed method. One module measures the distance by means of the SDS-TWR method (NanoLOC), and the other calculates the distance using only the RSSI (received signal strength indicator) value (Xbee). Xbee requires additional computation to convert the signal strength to distance. NanoLOC measures the distance using the difference of signal transmission time. Figure 8 shows the accuracy of the distances measured with the two types of modules. The horizontal axis represents the actual distance and the vertical axis displays the distance measured by the modules. Xbee shows a larger error than NanoLOC, as the RSSI signal is susceptible to interference from other signals. The accuracy of NanoLOC is much better than that of Xbee even though it does not use additional filtering operations.

4.2. Analysis of Factors Influencing Sensor Placement. For accurate localization with minimal cost, we determine the factors that affect the accuracy of localization and evaluate how much each factor influences the accuracy. We select the factors as the sampling rate of the fixed node and the interval between consecutive nodes.

The first factor to have a strong influence on localization accuracy is the sampling rate of the node. In order to measure the accuracy with respect to the sampling rate, we set the sampling rate as manipulated factor and other variables as controlled factors. Specifically, we fix the interval of nodes as 20 m and the number of nodes is not considered since it does not affect sampling rate.

Table 1 shows the accuracy of distance measurement and signal loss depending on the sampling rate. We analyze 300, 600, 1200, 1500, and 3000 output data while changing the sampling rate to 10, 20, 40, 50, and 100 Hz during the 30 seconds. Signal loss is the percentage of lost data from the total sampled data at specific sampling rate. Accuracy is the ratio of error between the actual distance and the measured distance including lost data. In these experiments, maximum error bound is set to within ± 1 m considering walking speed 4~5 km/h [26]. As described in Table 1, the accuracy at

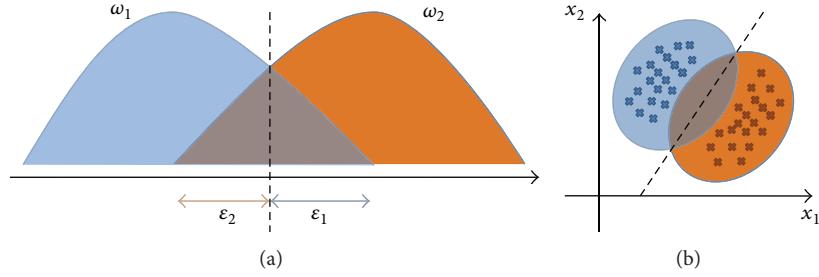


FIGURE 6: Decision boundary for estimating minimum Bayes error in one-dimension (a) and two-dimension with a linear decision boundary when $\Sigma_1 = \Sigma_2$ (b).

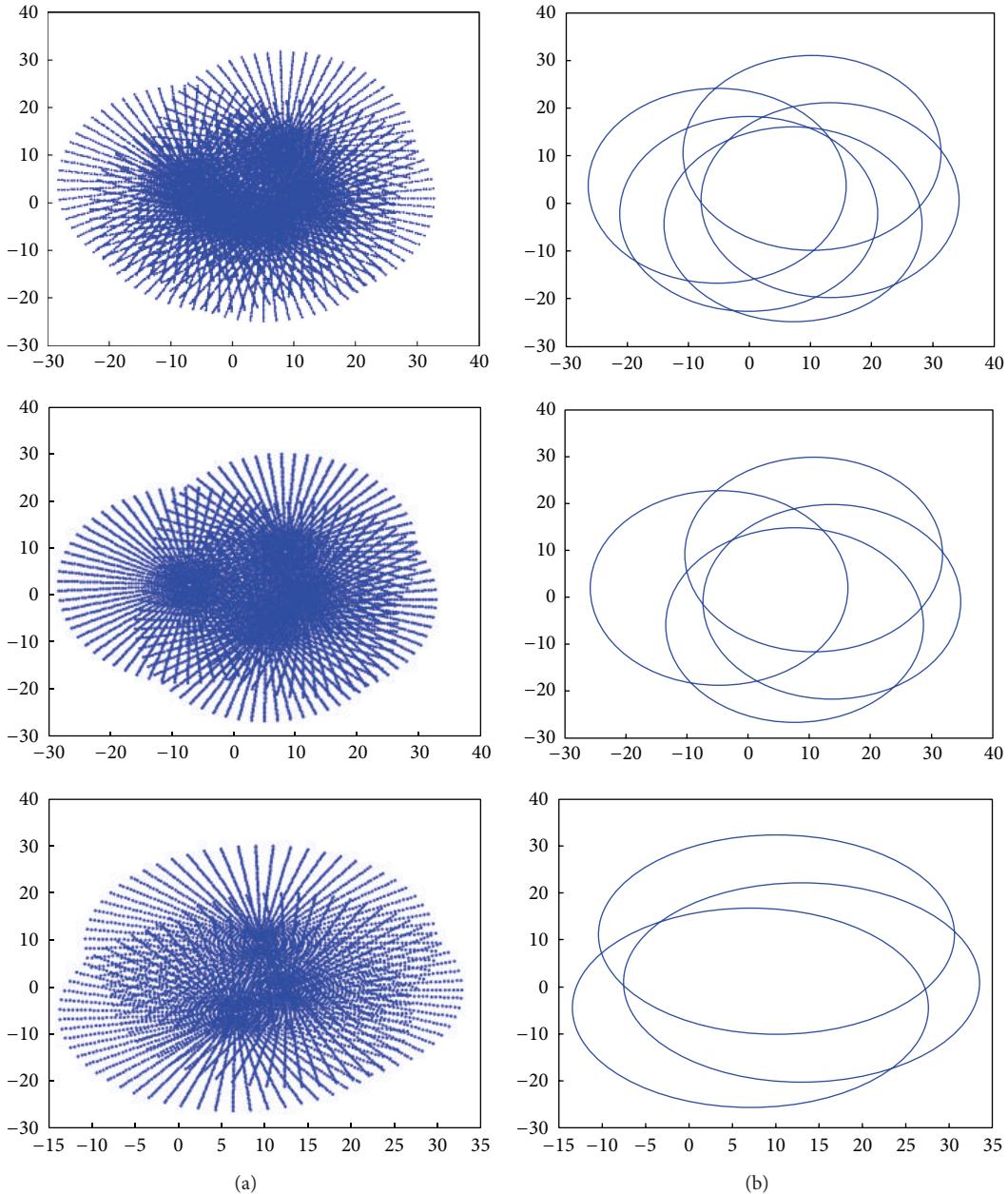


FIGURE 7: An example of optimal nodes selections for trilateration: (a) the signal points of nodes and (b) the signal range boundary of nodes.

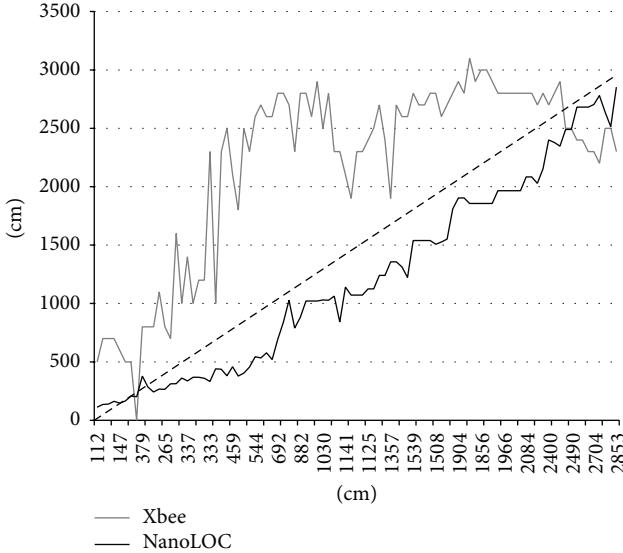


FIGURE 8: Distance accuracy comparison between the two modules: the horizontal axis represents the measured distance by the module and the vertical axis represents the actual distance.

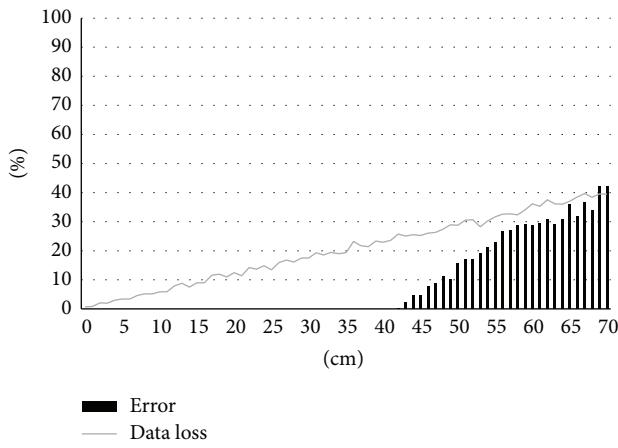


FIGURE 9: Error and data loss rates according to the distance between nodes.

sampling rates from 10 Hz to 40 Hz is proportional to the amount of received data without being affected by the signal loss. In addition, the accuracy at sampling rates from 40 Hz to 100 Hz decreases as the excessive increase of data generates a bottleneck and rapidly increases the signal loss. When the amount of received data increases, accuracy increases in general. However, when the sampling rate exceeds a specific value, signal loss increases rapidly and accuracy reduces. We determined the optimal sampling rate to be 40 Hz.

Determining the distance between nodes is very important for the optimal placement. In these experiments, the distance between nodes is selected as the second factor for evaluation. In order to determine the signal coverage of the module, the distance between nodes is set to manipulated

factor and the sampling rate of node is set to controlled factor which is determined by previous experiments.

To evaluate the accuracy with respect to the distance between nodes, we measure the data while arranging the node at regular intervals in a location where the interference of other signals is minimized. The error rate in this experiment is the percentage of erroneous data to total received data. We regard a data as an error when it is greater than 1 m which is tolerable distance in consideration of walking speed. Rate of signal loss is the percentage of lost data during a span of time.

Figure 9 shows the error and loss rates of the received data with respect to the distance between the two nodes. We analyze the result for 1200 packets acquired from one node installed per unit distance, which is measured for 30 seconds at 40 Hz (optimal sampling rate determined in the previous experiment) in simulation. The results show that there are no error packets until the distance becomes greater than 40 m, as shown in Figure 10. Furthermore, the results show that the data loss linearly increases with the increase of the distance between the two nodes. We determine that optimal distance between the two nodes is about 40 m, which is the maximum distance without erroneous data with minimum cost. The data loss rate at 40 m is about 25%, but this does not influence the accuracy of trilateration considering the sampling rate and walking speed.

We select two factors to analyze the characteristics of range-based ZigBee module by performing a heuristic approach based on statistical analysis. We can find the optimal placement with experiments on the basis of these two factors even if the ZigBee is changed.

4.3. Optimal Placement Using Proposed Method. To find the optimal placement of nodes, we virtually arrange them with the determined factors which are described in the previous subsection. The experiment is performed in a 600×500 m area that includes a pedestrian road, as shown in Figure 10. We assign 52 POIs (red points) and 9500 candidate locations where the nodes can be replaced.

Table 2 shows the results of cost savings and rate of unreach area in accordance with MCMB. The cost saving means the percentage of optimal nodes in total nodes (9500). The rate of unreach area represents the percentage of the area where the signal has not reached. The number of optimal nodes is determined according to MCMB. We arrange 9500 nodes in order to cover entire region.

As shown in Table 2, the minimum number of nodes for trilateration is increased as MCMB is increased. When MCMB is more than 2.0, the number of optimal nodes rapidly grows and cost saving is significantly reduced. The unreach area rate has decreased by increasing MCMB. If MCMB is more than 2.0, rate of unreach area becomes less than 1%. Based on the experimental results, we determine optimal MCMB as 2.0 in consideration of the cost and unreach area rate.

To verify the availability, we implemented an overlapping area estimation method and compared it with our method. The overlapping area method selects the minimum number of nodes required for trilateration by sequentially removing

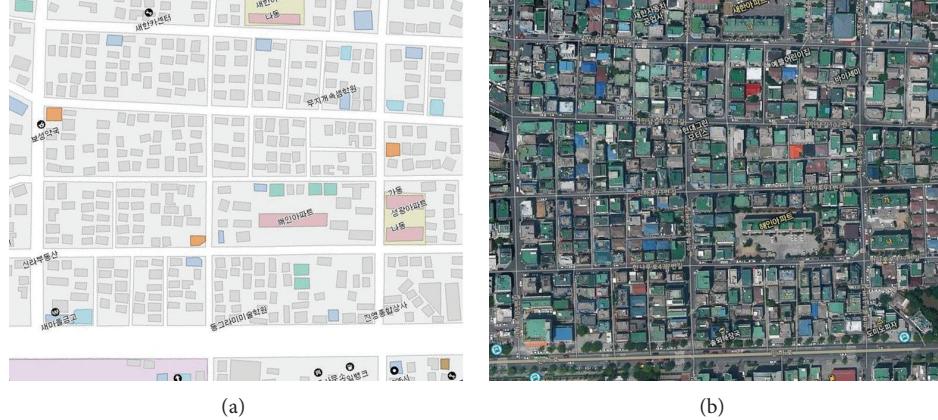


FIGURE 10: Example of testbed: 2D map (a) and LANDSAT image (b).

TABLE 2: Cost saving and rate of unreached area dependence on MCMB.

MCMB	Number of optimal nodes	Cost saving (%)	Unreached area rate (%)
1.0	47	99.50	23.00
1.2	55	99.42	13.21
1.4	61	99.35	10.89
1.6	68	99.28	8.37
1.8	71	99.25	5.83
2.0	76	99.20	0.17
2.2	219	97.69	0.10
2.4	589	93.80	0.08
2.6	1143	87.96	0.03
2.8	2340	75.36	0.02
3.0	4903	48.38	0.00

the node with the largest overlapping area. The sum of the overlapping area means the total area overlapped by other adjacent ZigBees, and an RN represents the node with maximum sum. In (9), $S_{(q,r)}$ represents the overlapping area of its neighbor, and $d_{(q,r)}$ is the Euclidean distance between two nodes q and r :

$$= r \left[2r \cos^{-1} \left(\frac{d_{(q,r)}}{2r} \right) - d_{(q,r)} \times \sin \left\{ \cos^{-1} \left(\frac{d_{(q,r)}}{2r} \right) \right\} \right]. \quad (9)$$

The optimal placement method consists of two steps. In *optimization* step, the proposed method determines an RNC (removable node candidate). In *verification* step, the RN is decided after we determine whether the RNC is required for trilateration (see Algorithm 1).

Table 3 shows the results of the two methods. The minimum number of nodes for trilateration was determined to be 76 nodes by our method and 112 by the overlapping area estimation method. The cost saved by the proposed method is up to 99.20% and the cost saved by overlapping area method

is 98.82% at most. In addition, the unreached area rate shows that the proposed method is 0.17% and overlapping area method is as high as 6.74%.

Figure 11 shows the optimal placement of nodes using both methods. The placement of nodes that is determined by the proposed method is displayed as black points and the arrangement of nodes using the overlapping area method is indicated as both the white and black points. The overlapping area method requires more nodes and shows high unreached area rate in comparison to the proposed method, as shown in Figure 11. Therefore, the proposed method can achieve an efficient localization over an area with a smaller number of nodes as well as save significant amounts of cost and time.

4.4. Localization Success Rate According to Interference. We have evaluated success rate of localization according to the rate of interference signals such as Wi-Fi. We measure success rate of arriving at the destination with four persons by using the proposed method. The interference rate is measured to relative interference strength by the number of access points (APs) of the Wi-Fi with the same frequency [27].

As shown in Table 4, the success rate is lower than 8% in places such as building with many Wi-Fi or other interference

TABLE 3: Optimal number of ZigBee nodes using two methods.

Estimation method	Number of ZigBee nodes		Cost saving (%)	Unreached area rate (%)
	Initial nodes	Optimal nodes		
Bayes error	9500	76	99.20	0.17
Overlapping area		112	98.82	6.74

TABLE 4: Experiments' relative comparison of success rate according to interference signal.

Interference rate (%)	Success rate (%)			
	User 1	User 2	User 3	User 4
10	98	95	95	94
30	69	65	65	63
50	42	35	39	36
70	13	10	16	14
90	5	4	8	9

```

Function optimization(Method)
  Input the method that will select to remove a ZigBee node
  Output an image that indicates the optimal ZigBee placement
  while stopFlag = true do
    if Method = BayesError then
      sortedList ← sortByBayesError()
    if Method = OverlappingArea then
      sortedList ← sortByOverlapArea()
    for all elements in the sortedList do
      stopFlag ← true
      if verification(index, sortedList) then
        remove the indexed node
        flag ← false
        break
      save all nodes in the output image
    end
  Function verification(index, sortedList)
    Input index: criterion node, sortedList: all nodes' List
    Output true/false value if remaining ZigBees are verified localizationof all positions
      after the indexed node is removed
    threshold ← the maximum value that can reliably measure
      distance by ZigBee node
    tempNode ← indexed node
    returnValue ← true
    remove the indexed node in the sortedList
    for all position which can walk in the map do
      for all elements in the sortedList do
        d ← distance between the current position and node
        if d < threshold then
          count++
          if count ≥ 2 then break
        if count < 2 then
          returnValue ← false
          break
      add the tempNode in the sortedList
    return returnValue
  
```

ALGORITHM 1: Pseudocode for determining optimal nodes.

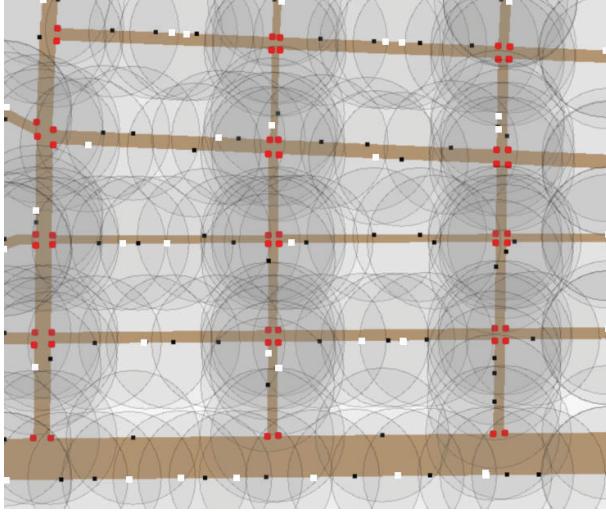


FIGURE 11: Optimal placement of ZigBee nodes using proposed method: POIs are presented as red points, the placements of nodes determined by the proposed method are displayed as black points, and the arrangement of nodes determined by the overlapping area method is indicated as white and black points.

signal. In spots such as parks and playgrounds, the success rate presents as higher than 95% since other interference signal is less than 10% in the place. As a result, users can arrive at the destination if there is rarely an interference signal having same frequency band.

5. Conclusions

We proposed a localization method using the range-based ZigBee node. The ZigBee nodes were arranged at known places, and the position of a user with a mobile base station was accurately estimated by means of trilateration. In addition, we proposed a method of optimal placement of fixed nodes by applying Bayes error estimation based on Gaussian distributions. We verified the effectiveness of the proposed method that estimates the position of a user with high accuracy using a minimum number of fixed nodes.

Conflict of Interests

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

Acknowledgments

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (no. 2012M3C4A7032781). This work was supported by the Inha University Research Grant.

References

- [1] J. H. Lee, K. Kim, S. C. Lee, and B. S. Shin, "Smart backpack for visually impaired person," in *Proceedings of the International Conference on ICT for Smart Society (ICISS '13)*, pp. 1–4, 2013.
- [2] G. Sun, J. Chen, W. Guo, and K. J. R. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 12–23, 2005.
- [3] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 41–53, 2005.
- [4] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, 2005.
- [5] S. Gezici, Z. Tian, G. B. Giannakis et al., "Localization via ultra-wideband radios: a look at positioning aspects of future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, 2005.
- [6] M. A. Youssef, A. Agrawala, and A. U. Shankar, "WLAN location determination via clustering and probability distributions," in *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom '03)*, pp. 143–150, March 2003.
- [7] D. Agrawal and Q. A. Zeng, *Introduction to Wireless and Mobile Systems*, Cengage Learning, 2010.
- [8] D. L. Lee and Q. Chen, "A model-based WiFi localization method," in *International Conference on Scalable Information Systems*, pp. 40:1–40:7, 2007.
- [9] J. Biswas and M. Veloso, "WiFi localization and navigation for autonomous indoor mobile robots," in *Proceeding of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 4379–4384, Anchorage, Alaska, USA, May 2010.
- [10] V. M. Olivera, J. M. C. Plaza, and O. S. Serrano, "WiFi localization methods for autonomous robots," *Robotica*, vol. 24, no. 4, pp. 455–461, 2006.
- [11] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, *Indoor Localization System Using RSSI Measurement Of Wireless Sensor Network Based On Zigbee Standard*. Wireless Sensor Networks, ACTA Press, 2006.
- [12] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted centroid localization in Zigbee-based sensor networks," in *Proceedings of the IEEE International Symposium on Intelligent Signal Processing (WISP '07)*, pp. 1–6, Alcalá de Henares, Spain, October 2007.
- [13] F. Sottile, R. Giannantonio, M. A. Spirito, and F. L. Bellifemine, "Design, deployment and performance of a complete real-time ZigBee localization system," in *Proceedings of the 1st IFIP Wireless Days (WD '08)*, pp. 1–5, November 2008.
- [14] S. Krishnan, P. Sharma, Z. Guoping, and O. H. Woon, "A UWB based localization system for indoor robot navigation," in *Proceedings of the IEEE International Conference on Ultra-Wideband (ICUWB '07)*, pp. 77–82, September 2007.
- [15] Z. Cemin, M. Kuhn, B. Merkl, A. E. Fathy, and M. Mahfouz, "Accurate UWB indoor localization system utilizing time difference of arrival approach," in *Proceedings of the IEEE Radio and Wireless Symposium*, pp. 515–518, January 2006.
- [16] E. Trevisani and A. Vitaletti, "Cell-ID location technique, limits and benefits: an experimental study," in *Proceedings of the 6th*

- IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '04)*, pp. 51–60, December 2004.
- [17] D. C. K. Yuen and B. A. MacDonald, “Vision-based localization algorithm based on landmark matching, triangulation, reconstruction, and comparison,” *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 217–226, 2005.
 - [18] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, “Localization from connectivity in sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 961–974, 2004.
 - [19] J. Li, J. Li, L. Guo, and P. Wang, “Power-efficient node localization algorithm in wireless sensor networks,” in *Advanced Web and Network Technologies, and Applications*, vol. 3842 of *Lecture Notes in Computer Science*, pp. 420–430, 2006.
 - [20] M. Bshara, U. Orguner, F. Gustafsson, and L. van Biesen, “Fingerprinting localization in wireless networks based on received-signal-strength measurements: a case study on WiMAX networks,” *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 283–294, 2010.
 - [21] A. Taok, N. Kandil, S. Affes, and S. Georges, “Fingerprinting localization using ultra-wideband and neural networks,” in *Proceedings of the International Symposium on Signals, Systems and Electronics (ISSSE '07)*, pp. 529–532, Montreal, Canada, August 2007.
 - [22] L. J. Xing, L. Zhiwei, and F. C. P. Shin, “Symmetric double side two way ranging with unequal reply time,” in *Proceedings of the IEEE 66th Vehicular Technology Conference (VTC '07)*, pp. 1980–1983, October 2007.
 - [23] W. Kim, J. G. Lee, and G. I. Jee, “The interior-point method for an optimal treatment of bias in trilateration location,” *IEEE Transactions on Vehicular Technology*, vol. 55, no. 4, pp. 1291–1301, 2006.
 - [24] J. Rozman, “Grid-based map making using particle filters,” *Journal International Journal of Autonomic Computing*, vol. 1, no. 2, pp. 211–221, 2009.
 - [25] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, Boston, Mass, USA, 1990.
 - [26] T. P. Andriacchi, J. A. Ogle, and J. O. Galante, “Walking speed as a basis for normal and abnormal gait measurements,” *Journal of Biomechanics*, vol. 10, no. 4, pp. 261–268, 1977.
 - [27] M. Petrova, L. Wu, P. Mähönen, and J. Riihijärvi, “Interference measurements on performance degradation between colocated IEEE 802.11g/n and IEEE 802.15.4 networks,” in *Proceeding of the 6th International Conference on Networking (ICN '07)*, Martinique, France, April 2007.

Research Article

Cognitive Radio-Based Vehicular Ad Hoc and Sensor Networks

Mohammad Jalil Piran, Yongwoo Cho, Jihyeok Yun, Amjad Ali, and Doug Young Suh

Electronics and Radio Engineering Department, Kyung Hee University, 1732 Deogyeong-daero, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Republic of Korea

Correspondence should be addressed to Mohammad Jalil Piran; piran@khu.ac.kr

Received 4 February 2014; Revised 21 May 2014; Accepted 21 May 2014; Published 5 August 2014

Academic Editor: Feng Hong

Copyright © 2014 Mohammad Jalil Piran et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Current advancements in vehicular networking lead to amplifying the issue of spectrum scarcity. Cognitive radio (CR) has emerged as the key technology that enables flexible, efficient, and reliable spectrum exploiting by unlicensed networks to use licensed spectrum bands in an opportunistic manner to alleviate the spectrum scarcity issue. We have already proposed vehicular ad hoc and sensor networks (VASNET) as a new networking paradigm for vehicular communication by utilizing wireless sensor nodes in two mobile and stationary modes. The nodes are employed to sense vehicles' activity, for example, car crashes. Like other unlicensed networks, VASNET is supposed to operate over unlicensed spectrum bands. However, due to emergency nature of accident alarms in VASNET that must be received in minimum period of time and avoiding packet loss, in this paper we investigate CR-based VASNET, named as CR-VASNET. Various challenges and issues are discussed in details. In order to save stationary nodes' power and enlarge the network lifetime, an optimization technique for relay node selection is proposed. Furthermore, for primary users (PU) protection, an energy detection-based scheme is suggested to calculate the probability of PU arrival. Our theoretical discussions and simulation results proved the efficient functionality of the proposed CR-VASNET as a promising vehicular networking paradigm.

1. Introduction

Wireless sensor networks (WSNs) are recently used in many civilian application areas including environment and habitat monitoring, healthcare applications, home automation, and vehicular traffic control [1]. As an application of WSN, vehicular networks received plenty of studies in the recent years. In [2] we have proposed a new networking paradigm for vehicular networks by employing wireless sensor nodes in an ad hoc topology. We entitled the proposed paradigm as vehicular ad hoc and sensor networks (VASNET). VASNET is a self-organizing ad hoc network comprised of a large number of sensor nodes. In VASNET there are two kinds of sensor node, some are embedded on the vehicles, called as vehicular nodes (VNs) and others are road side sensor (RSS), deployed at predetermined distances besides the highway road wherever communicating between VNs and infrastructure units is not possible. The VNs are used to sense the vehicle activity, accident detection for instance. We

can have some base stations (BS) such as Police Traffic Station, Firefighting Group, or Rescue Team.

In case of accident, the VNs estimate the accident severity by fusing the sensors readings, for example, collision, the jerk, inclination degree, and temperature. Then, the defuzzified result of the fused data as a decision making alarm will be routed to the BSs via RSSs. VASNET provides the capability of wireless communication between VNs and RSSs and increases safety, comfort, and information agility for vehicles on the highway roads.

VASNET as an application of WSNs is supposed to employ a fixed spectrum band allocation such as dedicated short-range communication band (75 MHz spectrum allocated in 5.9 GHz band), unlicensed band, for example, 2.4 GHz industry, science, and medicine (ISM) band, or a wideband using overlay techniques like ultrawideband (UWB) signaling. Due to increasing demands for wireless communications, efficient spectrum utilization comes into the picture. To address this challenge, cognitive radio has

emerged as a key technology, which enables VASNET to access both licensed and license-exempt spectrum bands (i.e., TV unused spectrum channels) in an opportunistic manner. The CR and VASNET share distinct similarity in the way they operate; sensing operations are carried out to collect information from the operating environment and respond accordingly. We call our proposed system as cognitive radio-based vehicular ad hoc and sensor networks (CR-VASNET). Our objective is to provide a clear picture of potential of CR-VASNET, the current state-of-the-art, the research challenges on this exciting topic, and propose practical methods to alleviate the issues.

Regarding the report of the authors in [3], around 70% or more of TV bands are unused in small cities and rural areas. Thus, the proposed CR-VASNET is particularly applicable for highway roads since most portions of a highway are in rural areas, where the spectrum band is quite clean and CR-VASNET is able to find significant unused spectrum channels. Furthermore, it is expected that the percentage is higher for highways passing through areas where few people live. Thus, in this paper, we focus on CR-VASNET utilizing TV white spaces on the highway roads in an opportunistic manner without harmful interference to the incumbents.

CR-VASNET composed of sensor nodes equipped with CR may gain benefit of the (1) dynamic spectrum access: the existing vehicular networks, for example, VANET deployments, supposed fixed spectrum allocation over very crowded license-exempt bands (ISM, for example). Unlicensed bands are used by many secondary users (SUs) such as IEEE802.11 hotspots, PDAs, and Bluetooth devices. However, VASNET must either operate over license-exempt bands or a spectrum lease for licensed band must be acquired. But, a spectrum lease for a licensed band increases the overall implementation cost. Therefore, for efficiency cooperation with other types of users, opportunistic spectrum access to the TV spectrum bands may be exploited in VASNET. (2) Access to multiple channels to adapt to various spectrum regulations: since the location is an important factor in TV bands availability and also with respect to this subject that regulation rules are different from region to region, thus, a particular spectrum band available in a region may not be available in adjacent regions. Since CR armed nodes are able to change their communication frequency, they would overcome the spectrum availability challenge as well. (3) Opportunistic channel access in case of dense traffic: upon the detection of an event, a large number of sensor nodes generate a dense traffic to send their readings to the base station. Collision, packet losses, downfall communication reliability, and high power consumption are the consequences of burst traffic. These abhorrent issues can be alleviated by opportunistic access to multiple alternative channels. (4) Optimum power consumption using adaptability: retransmissions and packet loss increase the power consumption in wireless networks due to their dynamic nature. CR equipped sensor nodes are able to adapt to varying channel conditions which result in optimum power consumption and efficient transmission.

Consequently, we define CR-VASNET as a distributed network of wireless CR sensor nodes—mobile and

stationary—which sense the vehicles' activities and collaboratively communicate their readings dynamically over available TV spectrum bands (TVWS) in multihop manner toward the base stations to satisfy application-specific necessities. CR-VASNET in vehicle-to-vehicle and vehicle-to-RSS nodes communication and even among vehicles' onboard diverse sensors will help improve energy efficiency, radio resource, traffic network management, accident incident avoidance and report, and so forth, by highway traffic awareness and route planning.

While the mentioned potential privileges and CR-VASNET definition introduce it as a promising paradigm for vehicular networks, the realization of CR-VASNET depends on addressing many serious issues, posed by the unique characteristics of WSNs, VANETs, and CRNs and further amplified by their union. Among many others, data fusion, localization of VNs, routing, inherent resource constraints of RSSs such as power and processing constraints, additional communication and processing demand imposed by CR capability, design of low cost and power-efficient CR sensor nodes, efficient opportunistic spectrum access, multihop, and collaborative communication over licensed and license-exempt spectrum bands are primary challenges to design and have real implementation of CR-VASNET. In this paper we delve some of the challenges and discuss them in details such as network architecture, node structure, spectrum management (spectrum sensing, spectrum analysis, spectrum decision, and spectrum mobility), and communication architecture. In order to reduce the number transmission and enlarge the network lifetime, we suggest a method to optimize the relay node selection. Furthermore, to overcome the most important issue of licensed channel access, an energy detection-based technique is suggested to protect the PUs.

The organization of the paper is as follows: we review the related works in Section 2. CR-VASNET architecture and relay node selection optimization scheme are discussed in Section 3. The structure of the nodes is explained in Section 4. In Section 5, the issues and challenges regarding dynamic spectrum management are discussed. The communication architecture is explained in Section 6. In Section 7, we present the simulation results. And finally, Section 8 concludes the paper.

2. Related Works

CR-VASNET inherits its challenges from three networking paradigms including cognitive radio networks (CRN), vehicular ad hoc networks (VANET), WSNs, and further union of them. CR technology as a promising solution for tackling the issue of spectrum scarcity received plenty of research works in recent years [4]. Existing research studies on CR focus on different problems including investigation techniques for spectrum management [5], MAC protocols [6], and routing [7]. Vehicular networking is prone to get benefit from CR technologies such as dynamic spectrum access [8–12]. The authors in [11] have presented a channel access management system to support QoS for data transmission in CR vehicular networks. They have formulated the channel selection

problem as a constrained Markov decision process (CMDP) model, where the PU traffic is slotted and the considered constraints include maximum probability of collision with the licensed users, maximum packet loss probability, and maximum packet delay for mobile nodes. Two types of channel, for example, shared-use and exclusive-use channels, were used for data transmission among vehicles. Hence, a CMDP for exclusive-use channel reservation and clustering control form a hierarchical MDP model. However, the proposed method is based on clustering scenario which causes large overhead; hence, it is not a noteworthy solution for vehicular networks according to their unstable topology.

In order to effectively minimize the cost of operation and ensure the resulting error in spectrum detection within the bounds, a framework for the placement of BSs, given the various capabilities of querying, geolocation, and local sensing capabilities that vehicles may be equipped with, has been proposed in [13]. The framework uses spectrum database as well as support from infrastructure. The guidelines for the uniform placement of stationary nodes have been provided in order to generate information about the availability of white spaces and thereby support the vehicular nodes. Although utilizing centralized approaches can effectively improve the reliability by improving data delivery, the centralized approach is not appropriate to support multihop communication. A machine-to-machine communication paradigm by utilizing CR technology coined as cognitive M2M and a coordination based energy-efficient spectrum detection protocol have been suggested in [14]. The main objective of the proposed paradigm is to enhance the flexibility, efficiency, and reliability of M2M communication, with the potential of CM2M for the smart grid in variant networks being presented. However, the authors did not thoroughly investigate the properties of large networks together with architecture designs and thus might overestimate the performance of M2M networks due to simplified assumptions.

Ghandour et al. in [15] proposed a system (named DCV) that provides data delivery guarantees using CRNs principles in congested VANETs. In fact, DCV detects data contention region in the network and if the contention delay exceeds the predetermined threshold, a central administration entity extends incrementally the spectrum allocated for the control channel using CRNs principles. Assuring of generation and transmission of data packets during the same interval of time were the main intention of the proposed system. The authors in [16] described a multiserver multipriority queuing model for the analysis of feasibility of vehicular dynamic spectrum access in TV white spaces. They resorted to CR technologies and studied the feasibility of dynamic spectrum access schemes that tapped into TV white spaces, and in an attempt to secure interleaved channels suitable for making communication between vehicles. In addition, they have utilized M/M/n and M/G/m models in order to evaluate the probability that a vehicle finds all channels busy as well as to derive the expected idle time. Two preemptive priority multiserver queuing models were formulated to represent only a snapshot of the system where static subset of the licensed channel happen to be interleaved. However, the model is built

on top of restrictive assumption that the number of intercommunicating vehicles was fixed. Indeed, when assuming vehicle-to-vehicle communication, it is highly unlikely that the number of communicating nodes is fixed.

CR is a promising and spectrum-efficient communication technology for resource constrained WSNs due to their event driven communication nature having bursty traffic depending on the characteristics of the event. WSNs may get benefit of CR to eliminate collision and excessive contention delay incurred by dense deployment of sensor nodes. Hence, recently, investigation of CR-based WSNs received plenty of research interests [17–21]. In [22] the main challenges and principles for delay-sensitive data transport in CRSN were introduced. Furthermore, the authors explored the existing transport protocols and algorithms devised for CRSN. The channel assignment problem in cluster-based multichannel CRSN has been investigated in [23]. They have developed an *R*-coefficient to estimate the predicted residual energy using sensor information and channel condition. Park et al. in [24] have investigated an optimal mode selection policy for CRSN powered RF energy harvesting. CRSN powered by RF energy harvesting has advantages of potentially perpetual lifetime and ability to solve interference problems. Maleki et al. in [25] considered a combined sleeping and censoring scheme as an energy efficient sensing technique for CRSN. They proposed a method to minimize the energy consumed in distributed sensing subject to constraints on the detection performance, by optimally choosing the sleeping and censoring design parameters.

However, the proposed solution for CR networks, CR sensor networks, and even CR-based VANETs cannot be directly applied to CR-VASNET. Indeed CR-VASNET has unique characteristics of WSNs, VANETs as well as CR and further amplified by their union. Thus, the existing solutions need to be customized according to the CR-VASNET traits including limited power of stationary nodes, high mobility of mobile nodes, dynamic topology, and frequent disconnections.

3. CR-VASNET Architecture

Mobile and stationary CR nodes form wireless communication architecture of CR-VASNET, over which the sensor readings obtained mostly by VNs are conveyed to the BSs via RSS nodes. The main duty of VNs is to perform sensing on the vehicles. In addition to the common sensing duty, CR-VASNET nodes (both VNs and RSS nodes) perform sensing on the TV spectrum bands in order to discover TVWS, too. VNs send their collected information in an opportunistic manner to their closest RSS node, and, subsequently, the RSS nodes retransmit the received data packet to the next RSS and ultimately to the base station. Figure 1 shows CR-VASNET topology and communications among nodes. The nodes can transmit their data over both licensed and unlicensed spectrum bands.

Generally, in a distributed sensor network, spectrum sensing may be performed by sensor nodes individually or cooperatively. Spectrum assignment can be based on the individual decision or a group of sensor nodes too. When

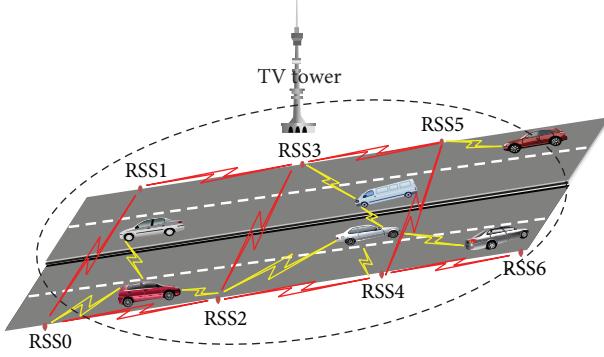


FIGURE 1: CR-VASNET topology and communications.

sensor nodes rely on their own spectrum sensing observation only, they may not be able to detect the appearance of primary users (PUs) due to shadowing. Achieving sensing in a distributed approach is known as cooperative sensing [26]. However, in CR-VASNET, the nodes share their spectrum sensing and decision result with the nodes in their vicinity.

CR-VASNET works on multihop mode; the number of hops which retransmit data packet should be minimized in order to minimize the consumption energy and subsequently enlarge the network life time. This can be achieved by sending data packets over longer distance by accompanying less number of relay nodes in transmission process. The communication range of the RSS nodes depends on their transmission power and the volume of the packet to be sent. Transmission over longer distances requires higher energy. The problem is to find maximum possible distance (nodes' transmission range) that sensor nodes can send data packets while taking care of the consumption energy. Hence, a node has to select another node which is so far from it but in its range. In this way the number of intermediate nodes is decreased as minimum as possible. To do that, we should consider transmission and receiving energy consumption as well as volume of the data packets. Suppose that n is the number of RSSs. The last node is represented as the BS. Let V_i be the total amount of the data in bytes, collected by node i , and P_i is the total power of the node. Let d_{ij} be the distance between node i and j , and let A be set of directed arcs between neighboring sensors. The power consumption to transmit from one RSS to another one depends on the distance between them and packet volume. The power required to communicate (send and receive) one bit data is

$$P_b = \left(P_{tx} + P_{rx} + \frac{P_{\text{encod}} + P_{\text{decod}}}{l} \right) d^\alpha, \quad (1)$$

where P_{tx} , P_{rx} , P_{encod} , and P_{decod} are transmission and receiving, encoding, and decoding power, respectively. It should be noted that P_{encod} is assumed to be negligibly small. α is the signal attenuation parameter in the range of $2 \sim 4$, with

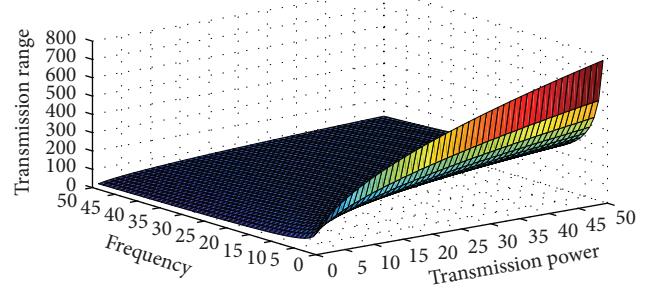


FIGURE 2: Transmission range versus frequency and transmission power.

$\alpha = 2$ corresponding to free space propagation, and l is packet payload. The transmission distance can be calculated as

$$d = \sqrt[{\alpha}]{\frac{(4\pi)^2 L_M R_N}{G_{tx} G_{rx} \lambda^2} \cdot \left(\frac{(h+l+t)}{R} \right)} \quad (2)$$

$$R_N = \frac{R_r}{R_o},$$

where L_M is the link margin compensating the hardware process variations and other additive background noise or interference, R_N is the receiver noise figure, R_r is the power spectral density of the total effective noise at the receive input, R_o is the signal-sided thermal noise power spectral density, G_{tx} is transmitter antenna gain, G_{rx} is receiver antenna gain, and λ is the wavelength of the transmitter signal in meters. While h is packet header length, t is packet trailer length, and R is data rate. The objective function is to maximize the transmission distance:

$$\begin{aligned} & \text{Maximize} \quad \sum_{\{i,j=n|(i,j) \in A\}} d_{ij} \\ & \text{Subject to:} \quad \sum_{\{j|(i,j) \in A\}} V_{ij} P_b + \sum_{\{j|(i,j) \in A\}} V_{ji} \leq P_i \\ & \quad \sum_{\{j|(i,j) \in A\}} V_{ij} P_b - \sum_{\{j|(i,j) \in A\}} V_{ij} \leq V_i \quad (3) \\ & \quad \sum_{\{j|(i,j) \in A\}} V_{ij} - \sum_{\{j|(i,j) \in A\}} V_{ji} \geq 0 \\ & \quad V_{ij} \geq 0. \end{aligned}$$

The constraints are as follows. (1) The data amount communicated (transmitted and received) by a node is limited by the node's power. (2) The difference of the data amount communicated at each node must be less than or equal to the amount of the node's readings. (3) The difference of incoming data amount and outgoing data amount must be greater or equal to zero. (4) The data amount should be positive. Hence, CR-VASNET nodes are able to increase their transmission range using lower frequency bands with the same transmission power because of better signal propagation characteristics.

As it has been illustrated in Figure 2, data transmission over longer distance improves lifetime and end-to-end delay,

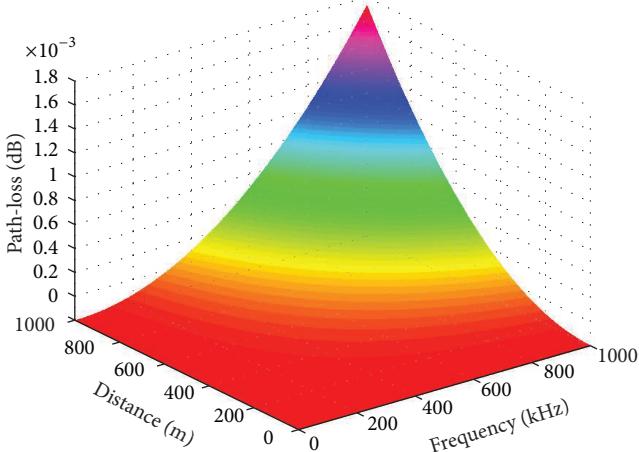


FIGURE 3: Path-loss rate increases due to high frequency and distance.

while it increases the path-loss. And also as shown in Figure 3, path-loss increases due to high frequency and longer distances.

Consider two nodes willing to communicate with each other in one-hop mode. The control data (e.g., available communication channel, suitable communication channels for data transmission, and location) and application data (e.g., alarm messages) due to their importance and possibly large volume should be transmitted with high reliability and efficiency, respectively. Therefore, we have to consider control channel disparate from the main channel. However, the control channels are opted from among relatively low frequency with longer range and lower data rate, and in contrary the main channel is picked up from high-frequency bands with higher data rate and shorter range. The elected channels may switch in response to the changes in link conditions dynamically. RSS nodes are aware of their location initially; however, VNs compute their location based on localization algorithm which we proposed in [27]. In addition, the sender and receiver share several common hash functions with location and time as the input key. Hence, the nodes are synchronized by reception of time and location pair. The hash functions are supposed to select channels from among lower frequency channels and return a number of proved channels. These channels are considered as candidate channels. Then, the sender and receiver sense the candidate channels and select the best one as the control channel. After selecting the control channel and exchanging the control data, the sender sends a list of appropriate channels for data transmission to the receiver. Then, the sender based on the received reply from the receiver determines data channel and starts data transmission.

4. CR-VASNET Node Structure

CR-VASNET sensor nodes are a low-cost and low-power device with the ability of sensing vehicle activity that is, velocity, collision, the jerk, inclination degree as well as

temperature. As illustrated in Figure 4, the main components of CR-VASNET sensor nodes are: sensing unit, processing unit, power unit and CR transceiver unit. The CR transceiver unit distinguishes CR-VASNET nodes from the classical sensor node, which enables the node to adapt transmission parameters dynamically. It is notable that RSS nodes like other traditional sensor nodes inherit the constraints in terms of processing and power resources, while VNs are involved with the high mobility issue.

Radio frequency (RF) front-end and the baseband signal processing unit are two major parts of a CR transceiver. To be adapted to the time variation of the RF environment, both radio front-end and the baseband signal processing unit can be configured by a control bus. In addition to sending and receiving, amplifying, mixing, and converting from analog to digital of the received signals are performed in the RF front-end, while modulation, demodulation, coding, and decoding are the tasks to be done in the baseband processing unit. Since the baseband processing unit of the CR is similar to the traditional transceivers, the novelty of CR is the RF front-end. The RF front-end is composed of (1) RF filter; by band-pass filtering the received RF signal selects the desired band. (2) The mixer mixes and converts the received signal with the generated RF frequency and to the baseband. (3) Voltage-controlled oscillator is composed of a signal with a known frequency for a given voltage to be mixed with the received signal. (4) Phase locked loop is for generating exact frequencies with acceptable resolution and also it guarantees that a signal is locked on a given frequency. (5) Channel selection filter is for capturing the requesting channel, and (6) automatic gain control keeps the gain or throughput power level of an amplifier constant over a range of input signal levels.

5. Dynamic Spectrum Management in CR-VASNET

The realization of CR-VASNET necessitates an efficient spectrum access management framework to handle the dynamic spectrum access of sensor nodes. Hence, to make the network protocols adaptive to the available TV spectrum bands, new functionalities are required. In the following subsections, we explore the main necessary functions for dynamic spectrum management and propose the appropriate spectrum access techniques.

5.1. Spectrum Sensing. In CR-VASNET, the nodes are able to operate over unlicensed bands as well as TV licensed spectrum bands. Nodes to capture licensed spectrum need to gather significant spectrum usage information prior to any transmission by sensing the spectrum bands. The spectrum sensing function by detecting TVWS gives the ability to the CR to adapt itself to the environment in which it operates. To detect the PUs, that is, TV users, that are receiving data within the SU's communication range is the most optimal way for spectrum TV white spaces detection. Spectrum sensing is considered as a physical layer issue. There are several spectrum sensing techniques such as cooperative detection,

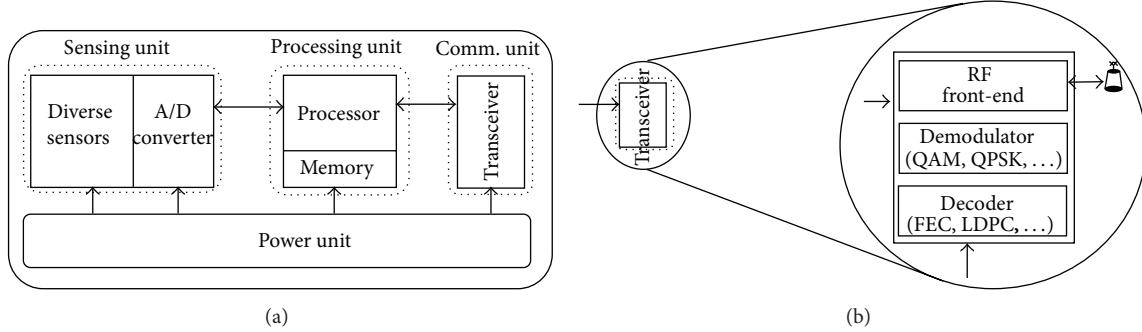


FIGURE 4: Anatomy of (a) wireless sensor node and (b) cognitive radio.

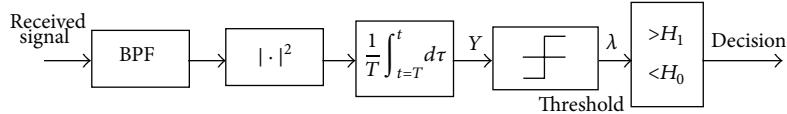


FIGURE 5: ED block diagram.

transmitter detection, and interference-based detection. The transmitter detection technique can be classified as matched filter detection, energy detection (ED), and cyclostationary feature detection. Regardless of what kind of detector, errors are unavoidable. Actually, the detector is sketched with the objective of bounding the error. There are two kinds of errors: (1) H_1 , an idle channel is detected as busy, which is known as false alarm too. It potentially leads to waste of transmission opportunity for the CR-VASNET users, and (2) H_0 , a busy channel is detected as idle, which is known as missed detection. It potentially leads to collision with TV users:

$$\begin{aligned} H_0 : Y &= w(k), \leq \lambda \\ H_1 : Y &= S(k) + W(k) \quad k = 1, \dots, n, \end{aligned} \quad (4)$$

where Y is the received signal, S is the signal that is going to be detected, W is the AWGN, k is the sample index, and λ is the threshold based on the receiver noise. $S(k)$ and $W(k)$ are zero-mean complex Gaussian random variables with variances $\sigma^2 w$ and $\sigma^2 s$ per dimension. When the value of S is equal to zero, it means there is no transmission going on from the PU. The most important and key metric in spectrum hole discovery is the probability of exact detection. However, it is not always guaranteed that the CR-users are able to detect correctly. In this regard normally two kind of errors may occur: (1) when a channel is idle, but it is detected as occupied, known as false alarm, and (2) when a channel is occupied, but it is detected as idle, misdetection.

Among the mentioned detection techniques, we choose ED technique [28] for our CR-VASNET to sense TV licensed bands in order to capture TVWSs. ED technique is an efficient and fast noncoherent technique that essentially computes a running average of the signal power over a window of prespecified spectrum length [29]. The advantage of the ED is due to its simplicity and ability to be applied on any type of deterministic signal and it requires the least

amount of computational power, and the receivers in this technique do not need any knowledge of the PUs' signals. In CR-VASNET, ED technique is applied in order to achieve efficient bandwidth. Since most of the TV spectrum bands are blank, the users in CR-VASNET performs ED technique for spectrum sensing to discover many interleaved spectrum bands, TV white spaces. CR-VASNET based on ED technique leads to increase safety and information agility for the vehicles on the highway roads affected by multipath shadowing and fading. Furthermore, data transmission of PUs is modeled as signal with a certain power, so the energy detector is optimal. Thereby, ED performance analysis in CR-VASNET with a variety of integrated techniques is of particular interest. CR-VASNET requires reliable and low latency wireless communication methods.

The signal is detected by comparing the output of the energy detector with a threshold based on the noise floor. Figure 5 shows the block diagram of the energy detector. The received signal is filtered with a band pass filter (BPF) to select the bandwidth. After that, the filtered signal is squared and integrated over the observation interval. The output will be compared to the threshold to determine the presence or absence of the PU. After analyzing the spectral environment, fast Fourier transform (FFT) based techniques will be applied to obtain frequency response. The detection test is based on the two hypotheses (4). The noise is supposed to be AWGN with zero mean and σ_w^2 . The probability of correct detection and false alarm [30] is associated with the value of λ ; the following equations show calculation of probability of detection and false alarm with the help of complete and incomplete gamma functions and Marcum Q-Function:

$$\begin{aligned} \text{Presence of signal if } \left(\sum_N (R_s[n])^2 \right) &> \lambda \\ \text{Absence of signal if } \left(\sum_N (R_s[n])^2 \right) &< \lambda \end{aligned}$$

$$\begin{aligned}
P_{\text{detection}} &= P \left\{ \left(\sum_N (R_s[n])^2 \right) > \lambda \mid H_1 \right\} \\
&= Q_M \left(\sqrt{2 \text{SNR}}, \sqrt{\lambda} \right) \\
P_{\text{False.alarm}} &= P \left\{ \left(\sum_N (R_s[n])^2 \right) > \lambda \mid H_0 \right\} = \frac{\Gamma(m, \lambda/2)}{\Gamma(m)} \\
P_{\text{detection}} &= Q \left(\frac{1}{\sqrt{2 \text{SNR} + 1}} \left(Q^{-1}(P_{\text{False.alarm}}) - \sqrt{N \cdot \text{SNR}} \right) \right), \\
\end{aligned} \tag{5}$$

where R_s is the received signal, N is number of samples, SNR is signal-to-noise ratio, and M is the Marcum Q -function factor. From the above-stated probability function of false alarm it can be observed that $P_{\text{False.alarm}}$ is independent of SNR; hence, under H_0 it means the presence of the PU. With the fading environment under Rayleigh fading, the result of $P_{\text{detection}}$ function will be the probability of detection given instantaneous SNR; thus, probability of PU detection is

$$\begin{aligned}
P_{\text{detection}} &= e^{-\lambda/2} \sum_{k=0}^{m-2} \frac{1}{k!} \left(\frac{\lambda}{2} \right)^2 + \left(\frac{1 + \text{SNR}}{\text{SNR}} \right)^{m-1} \\
&\times \left(e^{\lambda/2(1+\text{SNR})} - e^{-\lambda/e} \sum_{k=0}^{m-2} \left(\frac{\lambda \text{SNR}}{2(1+\text{SNR})} \right)^k \right). \\
\end{aligned} \tag{6}$$

5.2. Spectrum Analysis. The available TV white spaces show various characteristics which vary over location and time. Since CR-VASNET users are armed with the CR based physical layer, it is essential to understand the characteristics of different TV bands. Spectrum analysis as a part of spectrum management enables the characterization of various TV spectrum bands, which can be used to get the spectrum suitable to the user necessities. The sensed TVWS should be described considering (1) time varying radio environment, (2) spectrum band information, and (3) PUs' activity. The quality of a white space can be presented by some parameters such as holding time, path-loss, channel error rate, link layer delay, and interference level.

5.3. Spectrum Decision. CR-VASNET nodes must analyze the sensing information and make a proper decision about the channel and transmission parameter (such as carrier frequency range, modulation strategy, and maximum output power) for current transmission. Spectrum management function must be conscious about system quality of service (QoS) requirements too. Bandwidth of transmission, delay bound, acceptable error rate, transmission mode, and data rate could be some instances of QoS parameters. It has been shown that the TV spectrum sensing result will be the same in any given location [31]. Based on this observation, usually spectrum decision of the nodes for the adjacent nodes will be similar. Because of this, if nodes try to access a channel based on their individual spectrum decision, collisions cannot be dissembled. At the time of the collision, nodes try to switch

to another available channel, since all the nodes are running a similar algorithm; the collided nodes leave the previous channel and collide again on the new founded channel. As a result, we conclude that spectrum decision in CR-VASNET must be coordinated to achieve the maximum overall utilization and optimum power consumption.

In CR-VASNET, the users need to specify spectrum availability individually based on their observation. Since, the observed range of nodes is limited and less than their transmission range, if the nodes find the TV white spaces, then their transmission may cause interference at the TV receiver inside their transmission range. Thus, in CR-VASNET, spectrum sensing requires an efficient cooperation approach in order to prevent interference to TV users outside the observation range of each CR-VASNET user. In cooperative sensing, observations from multiple SUs are combined to improve detector performance. Particularly, cooperative sensing is useful since the performance of a single detector can severely degrade due to fading or shadowing. For cooperation, when a CR-VASNET user detects TV user activities, it has to notify its observation to the nodes in its vicinity to vacate the busy spectrum. However, the mentioned control channel (explained in Section 5.1) is used to discover neighbors of a CR-VASNET user and exchanging sensing information. However, in CR-VASNET coordination and spectrum decision is going to be handled by distributed approaches. In this approach, nodes share their yielded results of spectrum sensing and decision with their adjacent nodes. The salient advantages of the mentioned approach are minimum communication and power overhead.

5.4. Spectrum Mobility. CR-VASNET target is to utilize the TV spectrum in a dynamic fashion by allowing the radio terminals to transmit data in the best available TV spectrum band. However, the nodes have to obtain the best TV available spectrum. We define spectrum mobility as the process when a CR-VASNET node changes its operating frequency. In the following subsection, we explain the spectrum hand-off concept.

5.4.1. Spectrum Hand-Off. Detection of presence of PUs to avoid harmful interference is a serious challenge in CR-VASNET. However, in case of presence of PUs or, if current channel status becomes unsatisfactory for an operation, the CR-VASNET users must immediately switch to another available channel decided by an effective spectrum decision approach. The CR-VASNET users must do it compulsorily, even if they have an ongoing transmission. This function is known as "spectrum hand-off" [27]. When spectrum hand-off is required, an alternative TV white space should be determined. Then sender and receiver nodes have to handshake for transmission over new channel. After that, nodes can continue their transmissions. It is important for the mobility management algorithms to learn about the duration of a spectrum hand-off function in advance. This information should be prepared by the sensing mechanism. Once the mobility management protocols acquired the information about this latency, they have to guarantee that the ongoing transmission of the CR-VASNET user endures the least

performance degradation. Subsequently, multilayer mobility management protocols are necessary for accomplishing the spectrum management mechanisms. The mentioned protocols support mobility management adaptive to the various types of CR-VASNET applications. While spectrum hand-off function is going on, a TCP connection can be put to a wait state. However, it is needed to learn the new parameters and guarantee that the transmission from the previous parameters to new parameter is carried out quickly, because the TCP parameters will change after spectrum hand-off. Data packets in CR-VASNET are very vital and the system must take care to prevent any packet loss. But during hand-off mechanism it seems that probability of packet loss increases. In this case, the mobility management algorithms must deploy scenario to store the data packets, which are transmitted while spectrum hand-off is going on.

6. CR-VASNET Communication Architecture

The sensor nodes are deployed in vehicles as well as both sides of highway roads. Vehicular nodes have the capabilities to collect imperative data and route data to the base stations. Sensor readings are routed to the end users by multihop infrastructure architecture via intermediate nodes, that is, RSS nodes. The BSs may communicate with the task manager node via satellites or the Internet. The protocols stack which may be used by CR-VASNET nodes consists of five layers and three planes [32]. The planes are to help sensor node to coordinate the sensing tasks and lower overall power consumption. More specifically, the power management plane manages power consumption, for example, defining sleep and wake status for the nodes. The mobility management plane monitors the movement of sensor nodes, so a route back to the user is maintained always. And finally, the task manager plane balances and coordinates the sensing tasks given to a given region. In the following subsections, we investigate and briefly explain the specific design consideration of each communication layer of nodes with respect to dynamic spectrum management.

6.1. Physical Layer. The physical layer of sensor nodes—both VNs and RSS nodes—must provide the ability of reconfiguration of the operating frequency: channel coding, modulation, and also output power without applying any change in hardware structure. For reconfigure-ability of sensor nodes designing of software defined radio-based RF front-end transmitter and receiver is indispensable. Anyhow, there are many more challenges for designing a suitable physical layer for the nodes due to their resource constraints. In summary, we can mention the following as the most important issues: (1) wideband spectrum sensing, (2) cognitive learning, (3) advanced modulation schemes, (4) limited capabilities of A/D converter, (5) heavy signal processing algorithms, and (6) detecting weak signals, for example, presence of PUs.

6.2. Data Link Layer. Generally, this layer is responsible for reliable sending and receiving of data frames between communicating nodes. The most important functionalities

of data link layer are medium access control (MAC) and error control and correction. However, in CR-VASNET with respect to the dynamic spectrum management and energy efficiency, it tries to achieve the goals. Forward error correction (FEC) and automatic repeat request (ARQ) are the main error control schemes proposed for WSNs. In ARQ, high power consumption and low bandwidth utilization are the major weakness opposite to its advantages like its simplicity. However, it seems that FEC approaches are promising for CR-VASNET nodes due to their constraints. In FEC schemes, to recover bit errors, a particular magnitude of redundancy is included in the packet to be used by the receiver.

The complexity of the error correction algorithm and redundancy are effective on the amount of error that can be corrected. CR-VASNET is assumed to be able to access multiple TV frequencies. However, each TV channel may have various conditions, so a fixed FEC approach may not provide optimal results for different channels. It should be considered that, in some cases, when the channel condition is satisfactory, ARQ in comparison to FEC returns acceptable performance. Thus, this trade-off should be considered and it seems that combination of these two—FEC and ARQ—schemes may yield better performance. MAC protocols are supposed to provide the nodes with means to access the communication medium in efficient and fair manner. In CR-VASNET, data packets are very important, and all the capabilities must be employed to decrease packet loss as much as possible. Acting based on RTS/CTS (exchange of acknowledgment packets) is a proper way. Therefore, once an originated packet from VNs is received by the RSS nodes, to be forwarded to the BS, the RSS nodes may handshake to negotiate on the available channel prior to any transmission. Exchanging of acknowledgment packets is necessary for channel negotiation.

6.3. Network Layer. CR-VASNET as an application of WSN needs special multihop wireless routing algorithm among the vehicular nodes, RSS, and the base stations. An energy efficient and fair routing algorithm for vehicular sensor networks has been proposed by us already [33]. CR-VASNET inherits network layer issues from traditional WSNs [34, 35] and mobile ad hoc networks (MAENT) [36–38] such as infrastructure-less, unstable topology, multihop networking, energy efficiency data-centric routing, attribute-based routing, and localization, and rather than the mentioned challenges, the proposed routing protocols for CR-VASNET must consider dynamic spectrum access issues like channel access delay, interference, operating frequency, and bandwidth too. Furthermore, CR-VASNET as a multihop network, which the RSS nodes send data packets to the next hope, spectrum decision may change the next hope status. And also, because RSS nodes are limited in energy, hop count must be considered and it should be as less as possible. Hence, a new routing algorithm which takes care of both sensor network metrics and opportunistic spectrum access is required. Routing algorithms in ad hoc and sensor networks are classified as topology-based and position-based. The topology-based routings are categorized as proactive routing, reactive routing, and hybrid routing algorithms. Proactive

routing protocols are not fair for dynamic topology caused by opportunistic channel access in CR-VASNET. Therefore, it seems that reactive routing algorithms are more suitable. However, with the communication overhead and increased contention, dynamic spectrum aware reactive routing can be considered for CR-VASNET.

6.4. Transport Layer. End-to-end reliable delivery, congestion control to preserve scarce network resources, for example, energy, and taking care of application based QoS requirements are the major functionalities of transport layer. When sensor nodes detect an event, they try to send their readings towards the base station. This makes a bursty traffic into the network. Significant sensor collected data must be reliably delivered to base station to obtain detection and tracking of an event signal. Simultaneously, if the multihop network capacity exceeded, congestion is the result which thereby yields high power consumption. However, there is delicate balance between reliability and energy efficiency, which has been the main focus of transport layer solutions proposed for WSNs so far [3]. While the mentioned balance between reliability and energy efficiency is owned by CR-VASNET, the CR features bring other issues like varying channel characteristics and temporal inability to transmit silent sensing period. However, we can mention that spectrum aware reliability, opportunistic energy efficient transport, and real-time cognitive reliable transport are the main issues involved with this layer for CR-VASNET.

6.5. Application Layer. In case of WSNs, the application layer may be responsible for some functions like generation of information, interest and data dissemination, feature extraction of event signals, methods to query sensors, and data aggregation and fusion. However, in our proposed system, these services have to utilize the capabilities of CR-VASNET while conforming to its constraints. Therefore, designing of new protocols for this layer with respect to its pros and cons is required.

The purpose of data aggregation and fusion protocols is to reduce the communication cost while it tries to increase estimation performance. However, in CR-VASNET nodes' transceivers may tune to various channels; it means that a node is not capable of hearing all data transmission in its surrounding area. In [30] we have proposed a fuzzy-based data fusion protocol for vehicular sensor networks, in which we tried to employ fuzzy techniques to reduce number of transmissions or transmitted packet size to improve energy efficiency and network utilization. Nonetheless, still there are some considerable issues like developing approaches to do sampling of the event signal and collect sensory information based on spectrum availability. Specifically in CR-VASNET, where the sensing result should be sent to the base station when an event happens, mechanisms to discover the available spectrum in on-demand fashion must be investigated. From another view of CR-VASNET, where there is a query from the base stations, new query methods that take spectrum availability into account must be designed.

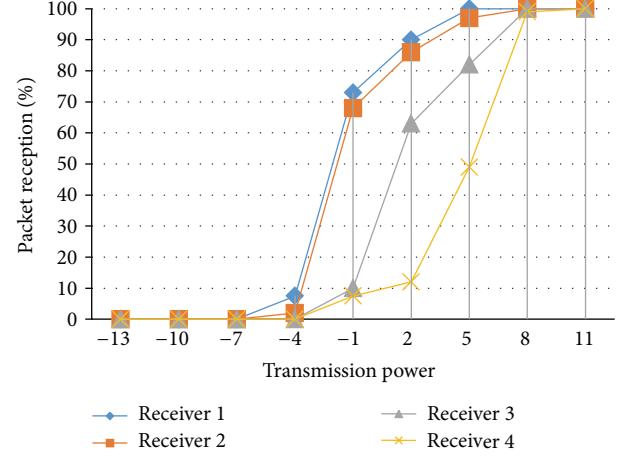


FIGURE 6: Percentage of successful packet reception versus transmission power over different distance between the sender and the receiver.

7. Simulation Results

In this section, we show our simulation results for distance optimization and PU detection. For simulation of distance optimization, we assumed that there are 100 nodes randomly distributed. One of them is supposed to be the final destination, the BSS. The transmission power is supposed to be in range of -13 to 11 dBm. The link distance between two adjacent nodes is varied between 15 and 30 meters. The radio dissipating is assumed to be 350 nJ/bytes to run the sender or receiver circuitry and 750 pJ/byte/m 2 for the sender amplifier. The other variables are assumed to be $\alpha = 2$, $L_M = 40$ dB, and $R_N = 10$ dB, and G_{tx} and G_{rx} are equal to 5 dBi. In order to observe the optimum distance, we measured link quality varying the one-hop receiver at relatively long distance. Suppose that there are four receiver nodes in the sender range but at the farthest distance from the sender. The distance among the receivers is varied between 1 and 3 meters. The variation in the link quality is very significant with the change of distance between the sender and the receiver. From the results shown in Figure 6, it can be observed that severe link quality variation can be expected with different distance from the sender. Our simulation results show that new reliable, communication links that do not exist at far distance from the sender at default transmission power can be generated with changing of transmission power.

For simulation of PU detection probability, we assumed that 100 RSSs are randomly deployed in area of 500 m \times 500 m area. Each RSS make a local binary decision, 1 for presence of the PU and 0 for absence of the PU. The RSS sum up the collected local binary decisions from its neighboring RSSs and compare with a threshold. If the sum is not less than the threshold, then RSS makes its binary decision as 1, and otherwise 0. We simulate the probability of detection under sensitivity to SNR with respect to the effect of shadowing and fading. Figure 7 shows probability of detection versus SNR curves over fading channels with different time bandwidth factor ($t = \{100, 750, 1500\}$), under

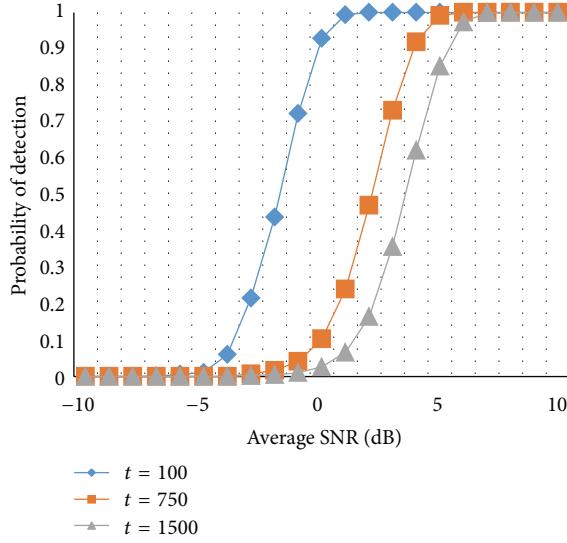


FIGURE 7: Probability of PU detection versus average signal-to-noise ratio.

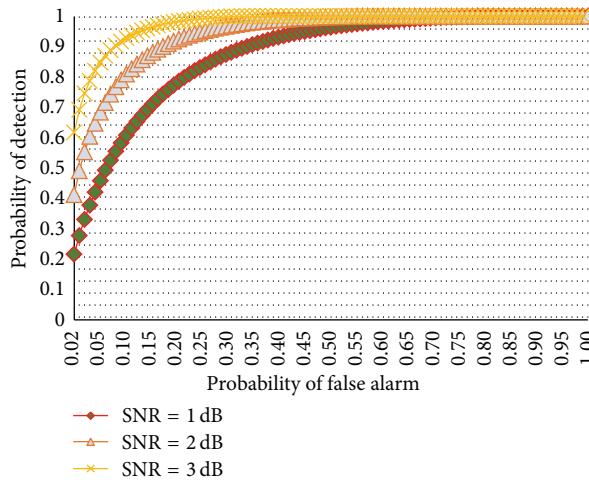


FIGURE 8: Probability of PU detection versus probability of false alarm with different SNR values.

AWGN channels. The results improve the protection of PUs by having high probability of PU's signal presence.

Figure 8 shows probability of detection and probability of false alarm with different SNR values, for example, 1, 2, and 3 dB. It can be observed that the probability of detection is increased, when the probability of false alarm increases.

8. Conclusion

In an opportunistic manner and adaptability to the channel condition, spectrum utilization and communication can be increased by employing CR Technology. The mentioned features are applicable to the vehicular sensor networks with respect to their unique constraints. Furthermore, CR is able to provide multiple channel availability, which can be exploited to overcome the issues caused by bursty traffic of

vehicular sensor networks. In this paper, we have investigated Cognitive Radio-based Vehicular Ad Hoc and Sensor Networks (CR-VASNET), as a new vehicular networking paradigm formed by adopting CR capabilities in Vehicular Ad Hoc and Sensor Networks to access TV white spaces in an opportunistic manner. We discussed advantages and constraints of CR-VASNET such as; network architecture, node structure, spectrum access management, and communication architecture. We have optimized the distance between the sender and the receiver (relay node selection) to save the RSSs' energy consumption and consequently enlarge the network lifetime. Furthermore we have proposed a scheme for PU detection in order to mitigate interference from CR-VASNET users to the incumbents. The results of our investigations and simulations introduce the proposed CR-VASNET as a potential candidate for vehicular networking to achieve higher safety and convenience in the highway roads.

Conflict of Interests

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

Acknowledgment

This research was funded by the MSIP (Ministry of Science, ICT and Future Planning), Korea, in the ICT R & D Program 2014.

References

- [1] I. F. Akyildiz, W. Su, Y. Sanakararubramanian, and E. Cayirci, "Wireless sensor networks: a survey," *IEEE Communication Magazine*, vol. 40, pp. 102–114, 2002.
- [2] G. R. Murthy and M. J. Piran, "Instantaneous Accident Detection and Notification System," 3780/CHE/2011 A, November, 2011.
- [3] B. Lennett, *Rural Broadband and the TV White Space: How Unlicensed Access to Vacant Television Channels Can Bring Affordable Wireless Broadband to Rural America*, New America Foundation: Wireless Future Program, Washington, DC, USA, 2008.
- [4] K. Ahmed, D. Perkins, and M. Bayoumi, "Opportunistic spectrum access challenges in distributed cognitive radio networks," in *Cognitive Radio Networks*, pp. 33–39, Springer, New York, NY, USA, 2013.
- [5] W.-Y. Lee and I. F. Akyildiz, "Spectrum-aware mobility management in cognitive radio cellular networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 4, pp. 529–542, 2012.
- [6] M. Timmers, S. Pollin, A. Dejonghe, L. van der Perre, and F. Catthoor, "A distributed multichannel MAC protocol for multihop cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 446–459, 2010.
- [7] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: challenges and solutions," *Ad Hoc Networks*, vol. 9, no. 3, pp. 228–248, 2011.
- [8] W. Kim, M. Gerla, S. Y. Oh, K. Lee, and A. Kassler, "CoRoute: a new cognitive anypath vehicular routing protocol," *Wireless Communications and Mobile Computing*, vol. 11, no. 12, pp. 1588–1602, 2011.

- [9] M. di Felice, R. Doost-Mohammady, K. R. Chowdhury, and L. Bononi, "Smart radios for smart vehicles: cognitive vehicular networks," *IEEE Vehicular Technology Magazine*, vol. 7, no. 2, pp. 26–33, 2012.
- [10] M. di Felice, K. R. Chowdhury, and L. Bononi, "Cooperative spectrum management in cognitive vehicular Ad Hoc networks," in *Proceedings of the IEEE Vehicular Networking Conference (VNC '11)*, pp. 47–54, November 2011.
- [11] D. Niyato, E. Hossain, and P. Wang, "Optimal channel access management with QoS support for cognitive vehicular networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 4, pp. 573–591, 2011.
- [12] M. Pan, P. Li, and Y. Fang, "Cooperative communication aware link scheduling for cognitive vehicular networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 760–768, 2012.
- [13] R. Doost-Mohammady and K. R. Chowdhury, "Design of spectrum database assisted cognitive radio vehicular networks," in *Proceedings of the 7th IEEE International ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM '12)*, 2012.
- [14] Y. Zhang, R. Yu, M. Nekovee, Y. Liu, S. Xie, and S. Gjessing, "Cognitive machine-to-machine communications: visions and potentials for the smart grid," *IEEE Network*, vol. 26, no. 3, pp. 6–13, 2012.
- [15] A. J. Ghandour, K. Fawaz, and H. Artail, "Data delivery guarantees in congested vehicular ad hoc networks using cognitive networks," in *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC '11)*, pp. 871–876, Istanbul, Turkey, July 2011.
- [16] S. Chen, A. M. Wyglinski, S. Pagadarai, R. Vuyyuru, and O. Altintas, "Feasibility analysis of vehicular dynamic spectrum access via queueing theory model," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 156–163, 2011.
- [17] J. A. Han, W. S. Jeon, and D. G. Jeong, "Energy-efficient channel management scheme for cognitive radio sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1905–1910, 2011.
- [18] Z. Liang, S. Feng, D. Zhao, and X. S. Shen, "Delay performance analysis for supporting real-time traffic in a cognitive radio sensor network," *IEEE Transactions on Wireless Communications*, vol. 10, no. 1, pp. 325–335, 2011.
- [19] C.-M. Wong and W.-P. Hsu, "Short paper: study on cognitive radio in IEEE 802.15.4 wireless sensor networks," in *Proceedings of the IEEE World Forum on Internet of Things (WF-IoT '14)*, pp. 179–180, Seoul, Republic of Korea, March 2014.
- [20] A. O. Bicen and O. B. Akan, "Reliability and congestion control in cognitive radio sensor networks," *Ad Hoc Networks*, vol. 9, no. 7, pp. 1154–1164, 2011.
- [21] M. C. Oto and O. B. Akan, "Energy-efficient packet size optimization for cognitive radio sensor networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 4, pp. 1544–1553, 2012.
- [22] A. O. Bicen, V. C. Gungor, and O. B. Akan, "Delay-sensitive and multimedia communication in cognitive radio sensor networks," *Ad Hoc Networks*, vol. 10, no. 5, pp. 816–830, 2012.
- [23] X. Li, D. Wang, J. McNair, and J. Chen, "Residual energy aware channel assignment in cognitive radio sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 398–403, March 2011.
- [24] S. Park, J. Heo, B. Kim, W. Chung, H. Wang, and D. Hong, "Optimal mode selection for cognitive radio sensor networks with RF energy harvesting," in *Proceedings of the IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '12)*, pp. 2155–2159, Sydney, Australia, September 2012.
- [25] S. Maleki, A. Pandharipande, and G. Leus, "Energy-efficient distributed spectrum sensing for cognitive sensor networks," *IEEE Sensors Journal*, vol. 11, no. 3, pp. 565–573, 2011.
- [26] J. Zhao, H. Zheng, and G. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proceeding of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 259–268, Baltimore, Md, USA, November 2005.
- [27] J. Zhao, H. Zheng, and G. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 259–268, November 2005.
- [28] H. Rasheed and N. Rajatheva, "Spectrum sensing for cognitive vehicular networks over composite fading," *International Journal of Vehicular Technology*, vol. 2011, Article ID 630467, 9 pages, 2011.
- [29] "IEE Trial-use standard for Wireless Access in Vehicular Environment (WAVE)—Multi-channel operation," IEEE Std 1609.4-2006, 2006.
- [30] E. Peh and Y. Liang, "Optimization for cooperative sensing in cognitive radio networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 27–32, Hong Kong, China, March 2007.
- [31] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, 2004.
- [32] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [33] M. Jalil Piran and G. Rama Murthy, "A novel routing algorithm for vehicular sensor networks," *Wireless Sensor Network*, vol. 2, no. 12, pp. 919–923, 2010.
- [34] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proceedings of the IEEE*, vol. 55, no. 4, pp. 523–531, 1967.
- [35] C. Xin, B. Xie, and C. Shen, "A novel layered graph model for topology formation and routing in dynamic spectrum access networks," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 308–317, November 2005.
- [36] S. Alrabaee, A. Agarwal, N. Goel, M. Zaman, and M. Khawneh, "Routing management algorithm based on spectrum trading and spectrum competition in cognitive radio networks," in *Proceedings of the 7th International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA '12)*, pp. 614–619, Victoria, Canada, November 2012.
- [37] S. Krishnamurthy, M. Thoppian, S. Venkatesan, and R. Prakash, "Control channel based MAC-layer configuration, routing and situation awareness for cognitive radio networks," in *Proceedings of the Military Communications Conference (MILCOM '05)*, October 2005.
- [38] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, "A survey of transport protocols for wireless sensor networks," *IEEE Network*, vol. 20, no. 3, pp. 34–40, 2006.

Research Article

Dynamic Groups Based Adaptive DTN Routing Algorithms in Social Networks

Jixing Xu, Jianbo Li, Lei You, and Chenqu Dai

Information Engineering College, Qingdao University, Ningxia Road 308, Qingdao, Shandong 266071, China

Correspondence should be addressed to Jianbo Li; lijianboqdu@gmail.com

Received 15 January 2014; Revised 27 March 2014; Accepted 29 March 2014; Published 29 April 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Jixing Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the nonexistence of end-to-end path between the sender and the receiver in delay tolerant networks and mobile social networks, consequently successful message transmission faces great challenges. In this paper, an adaptive routing algorithm taking full use of gregariousness characteristics of moving nodes is proposed. We first abstract all social relationships and uniformly represent them using friendship. Then by dynamically dividing nodes into different social groups, we finish flooding message among the target social group where destination node resides. In addition, we propose a social group based flooding model and a message redundancy control model to select fewer but better relay nodes and further reduce message redundancy. Extensive simulations have been conducted based on the synthetic traces generated by working day movement model and the results show that the proposed routing algorithm can get a higher message delivery ratio and a lower overhead ratio compared to Bubble Rap, Epidemic, and ProPHET, thus proving a better routing performance.

1. Introduction

TCP/IP protocol is based on certain assumptions, such as continuous end-to-end connections, low delivery latency, symmetrical bidirectional data transfer rate, and low error rate. In this case, TCP/IP protocol is able to shield heterogeneous networks. So with the TCP/IP protocol stack, traditional Internet has achieved a great success. However in recent years, some emerging challenged networks deployed in extreme environment are unable to meet the above assumptions, for example, vehicular ad hoc networks [1], pocket switched networks [2, 3], underwater sensor networks [4], interplanetary internet, mobile social networks [5, 6], and so forth. These special networks are characterized by intermittent connectivity, sparse node density, limited network resource, node mobility, and so forth. In these networks, there may never be a complete end-to-end path between the sender and the receiver, so traditional TCP/IP protocol is difficult to get efficient achievements. Consequently successful message delivery in such networks faces great challenges.

In 2003, Kevin Fall first put forward the concept of delay tolerant networks (DTNs). Soon afterwards, the DTN architecture [7–9] is proposed, which introduces a bundle

layer between the application layer and the transport layer. With the bundle layer, DTNs can shield heterogeneous networks, thus communicating across multiple regions that have different types of network architectures and protocols. In order to cope with frequent network topology partitions and finish end-to-end data communications, DTN routing adopts the store-carry and forward strategy to relay messages hop by hop.

So far, a large number of routing protocols have been proposed to optimize the next hop relay node selection in the case of lacking of global topology knowledge. In order to select appropriate next hop relay nodes to gradually move closer to the final destination node so as to improve message delivery ratio, most of the proposed algorithms forward or replicate messages by taking into account nodes' physical attributes such as link status and geographic location. However, for specific DTN application scenarios, there usually are some more useful characteristics or information, and making good use of them can further improve routing performance, for instance, the DTN application scenarios in social networks, in which nodes are not completely isolated. On the contrary, a node usually has social relationships or social ties with other nodes. For a university, the students

in the same class have strong relationships that we call classmates. In this case, such students will encounter each other much more frequently than the students outside of the class. Another example is that people having similar interests will regularly get together and eventually form a stable group. In other words, nodes in social networks are characterized by gregariousness. This gives us a heuristic that we can first forward message to the target social group where destination node resides and then flood message among the group in order to quickly and successfully finish message transmission.

In this paper, taking full use of the gregariousness characteristics of nodes in social networks, we proposed dynamic groups based adaptive DTN routing algorithm (DGA). For the reason that nodes in social networks have a variety of social relationships and common interests, we first abstract these social ties and then uniformly represent them using friendship. We do not care about the type of the relationship between two nodes. But we seriously focus on the strength of the relationship, which is usually reflected by the encounter probability between the two nodes. According to nodes' social circles and similarities, we dynamically add nodes into different social groups and then implement the social group based flooding routing. Our main contributions are summarized as follows.

- (i) We abstract all social relationships and uniformly represent them using friendship. And then we use encounter probability to measure the strength of the friendship between two nodes.
- (ii) We first propose an ego group model to describe the friend circle of a certain node. And then based on the ego group, we add the node into a certain social group.
- (iii) We propose a social group based flooding model to estimate encountered nodes, thus selecting fewer but better relay nodes so as to improve message delivery ratio.
- (iv) We also present a message redundancy control model to avoid generating unnecessary redundant message copies, thus further controlling message redundancy.

The rest of this paper is organized as follows. Section 2 discusses some related works. Section 3 gives detailed descriptions of the proposed algorithm. The performance evaluations and comparisons, as compared to Bubble Rap, Epidemic, and ProPHET, are presented in Section 4. Finally, Section 5 summarizes this paper.

2. Related Works

Currently, scholars have proposed a large number of DTN protocols to address the end-to-end communication problem. Firstly, we gave a brief description of the two typical routing strategies Epidemic [10] and ProPHET [11] for the reason that our routing idea is inspired by them. Secondly, we introduced some latest social-based routing protocols.

Epidemic is a typical flooding-based routing algorithm, which tries to replicate messages to all encountered nodes for the purpose of quickly delivering message to destination

node. In case of sufficient buffer resource and bandwidth, Epidemic may be able to get the best message delivery ratio. But in most DTN application scenarios, the buffer resource is extremely restricted. In this case, Epidemic is difficult to get satisfying overall routing performance. Despite this, we still think this flooding idea is acceptable, but the key issue is that Epidemic does not evaluate encountered nodes, thus leading to flooding message in the whole network. If we can control the flooding scope to a small group where the destination node resides, then we will get a better routing performance. Our routing idea comes in part from this.

ProPHET is a typical history utility based routing algorithm, which takes full use of the encounter history information and transitivity to predict delivery possibility. There are three delivery predictability metrics in ProPHET. The first metric is defined in (1), where $P_{(a,b)}$ is used to predict and update the delivery predictability that node a can successfully deliver message to destination node b and $P_{\text{init}} \in [0, 1]$ is an initialization constant. The second metric defined in (2) is to update the delivery predictability as time progresses, in which $\gamma \in [0, 1]$ is the aging constant and k is the number of time units that have elapsed since the last time the metric was aged. The last metric in (3) uses the transitive property to update the delivery predictability, where $\beta \in [0, 1]$ is a scaling constant. In social networks, ProPHET is able to describe the case that the nodes with strong relationships will have higher encounter probabilities than strangers due to the higher encounter frequencies between them. This gives us the concept to abstract social relationships and the way to measure the strength of a certain relationship:

$$P_{(a,b)} = P_{(a,b)\text{old}} + (1 - P_{(a,b)\text{old}}) \times P_{\text{init}}, \quad (1)$$

$$P_{(a,b)} = P_{(a,b)\text{old}} \times \gamma^k, \quad (2)$$

$$P_{(a,c)} = P_{(a,c)\text{old}} + (1 - P_{(a,c)\text{old}}) \times P_{(a,b)} \times P_{(b,c)} \times \beta. \quad (3)$$

Bubble Rap [12] was the first proposed community based routing protocol, which relies on two social characteristics: community and centrality. Each node in the network belongs to at least one community and has a global rankness and a local rankness, which, respectively, represents the global popularity in the whole network and the local popularity in its local community. The delivery process of a certain message includes two phases. It first bubbles message up based on the global popularity until the message is delivered to the destination community to which the destination node belongs. Then in the local community, it bubbles the message up based on the local popularity until the message is delivered to its destination or its TTL is exhausted. In order to reduce the resource consumption, the original carrier deletes this message once the message has been successfully delivered to the destination community. The simulation results in this paper show that Bubble Rap gets certain advantages in delivery latency and average hop count.

Taking full use of human contact features, Wu and Wang propose a hypercube-based multipath social feature routing algorithm [13], which includes two unique phases: social feature extraction and multipath routing. Specifically,

in social feature extraction process, it first uses Shannon Entropy to capture the m most informative features to build social feature space and then transforms the routing problem into hypercube-based feature matching problem. In the second phase, it proposes two kinds of multipath forwarding strategies. The extensive simulations using the real datasets show that the proposed algorithm achieved a significant improvement in the message delivery ratio and the end-to-end delivery delay.

ComPAS [14] is a typical community-based routing protocol proposed by Xia et al. By exploiting social relationship while replicating message in community, ComPAS can significantly improve routing performance and achieve better efficiency and consistency while keeping the replica relocation cost as low as possible.

Xiao et al. propose a distributed optimal community-aware opportunistic routing (CAOR [15]). By using a home-aware model, they turn mobile social networks into a network that only includes community homes. Besides, they also prove that, in the network of community homes, the minimum expected delivery delay can be computed through a reverse Dijkstra algorithm. Based on the home-aware model, CAOR achieves a satisfying opportunistic routing performance.

HS [16] is a zero-knowledge assisted mobile social network routing algorithm proposed by Wu et al., which spreads a given number of message copies in an optimal strategy by theoretical analysis when assuming that the intermeeting times between any two nodes and between a node and a community follow exponential distribution. In addition, by constructing a Markov chain, HS can also calculate the expected delivery delay and derive an upper bound, thus getting a better routing performance.

3. Routing Framework

Before presenting our DGA routing algorithm, we first define two groups: Ego Group and Social Group in this section, and then introduce our social group based flooding model and message redundancy control model.

3.1. Friendship Definition. There are too many kinds of social relationships in social networks. Besides, nodes' social behaviors driven by common interests are also complicated. In this case, it may be very difficult to analyze and capture these relationships. In order to reduce routing complexity, we extract their common key features: a pair of nodes with certain social relationship or specific common interest will encounter each other more frequently than strange nodes. Moreover, the strength of their relationship can also be accurately measured by their encounter probabilities. In other words, frequent communications between two nodes indicate higher encounter probability values, which can in turn prove the existence of a strong relationship between two nodes. The key issue we are concerned with is the strength of the relationship, rather than the type of the relationship.

Consequently, we abstract all social relationships and uniformly represent them using friendship. Furthermore, we use encounter probability to measure relationship strength.

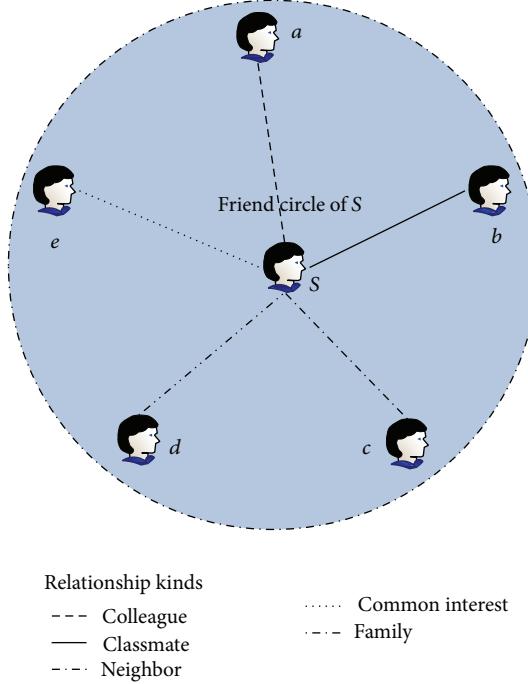
In this paper, we use (4) to measure the friendship degree and estimate whether two nodes could become friends, where $P_{\text{friend}} \in (0, 1)$ is an initialization constant representing the threshold to set up a friendship and can be set to an appropriate value according to the specific application scenario, and $P_{(a,b)}$ is shown in (1). If $\text{Friend}_{(a,b)} > 0$, then a and b are friend nodes. Finally, a pair of friends is defined as a pair of nodes with a strong social relationship (i.e., a high encounter probability). Here, we only focus on the direct friendship between two nodes. Therefore, we no longer use transitivity to predict the encounter probability. But taking into consideration that the established friendships may be broken as time progresses, we still use the aging (2) to update the encounter probability. With the aging mechanism, two nodes are able to determine whether they are still friends in real time:

$$\text{Friend}_{(a,b)} = P_{(a,b)} - P_{\text{friend}}. \quad (4)$$

3.2. Ego Group Definition. In social networks, due to different social behaviors and interests, a node will have different friends. Under normal circumstances, most friends of a node are not each other's friends or even do not know each other. The only social tie is that they have a common friend for a certain reason. In this case, the friends of a certain node are usually not closely associated. However, these friends information are very useful and can be used to determine the social group (defined in next subsection) where the node resides. Here, we define an Ego Group to describe all the friends of a node. As shown in Figure 1, nodes $a-e$ become the friends of node S , respectively, for different reasons. Then we define the ego group of S as the node set that consists of the node S and all the friends of S (i.e., nodes $a-e$). Now for a special example, if there is another node D , which has an ego group that also contains the nodes $a-e$, then S and D should be assigned to the same social group because they have a similar friend circle.

We assume that each node in social networks has one and only one ego group, in which the node is the unique ego node (i.e., the owner of the ego group) and other nodes are all its friends. The special instance is the ego group only containing the ego node itself. For an ego group, we do not care about the relationships between the friends of the ego node, but we do care about the composition of the ego group. Using (4), a node can easily establish its ego group and maintain it. As time progresses, if a node is no longer the friend of the ego node, then it should be deleted from the ego group.

3.3. Social Group Definition. In social networks, due to complicated relationships and common interests, the behaviors of nodes are characterized by gregariousness. In other words, there is a node set, in which member nodes are closely associated with each other and encounter each other much more frequently than nodes outside of the node set. If we can find such a node set where destination node resides, we can flood message only among the node set in order to improve

FIGURE 1: The Ego Group of node S .

message delivery ratio in the case of controlling network overhead. In this paper, we call the above node set the social group. Note that ego group can only reflect that the ego node encounters its friends very frequently but cannot reflect the encounter frequencies between the friends of the ego node.

Now there is a key problem: how to dynamically establish these social groups. Usually, the similar nodes should be added into the same social group. Ego group can reflect the friend circle information of an ego node, which is very useful and can be used as the metric to estimate the similarities between nodes. In this paper, we use the ego group information of a node to determine the social group where the node resides. Concretely, grouping decisions can be made when one of the following cases occurs.

- (i) If the friend circles of two ego nodes are almost the same and they are also each other's friends, then the two nodes are assigned to the same social group.
- (ii) If most member nodes of a certain social group are the friends of a node, then the node should also be added to the social group.

The detailed establishment process of social group is presented in Algorithm 1. For the above two cases, we, respectively, define the threshold parameters T_{eg} and T_{sg} to test whether the current node meets these cases. Lines 1-2 and line 8 are testing the first case by using T_{eg} . Lines 3-5 and line 10 are testing the second case by using T_{sg} . The two parameters can be set to an appropriate value according to the specific application scenario, thus controlling the size of social group. By increasing their values, we can reduce the group size and get a more closely associated social group and vice versa. In this paper, taking into account that we abstract

Input:

ego group of node a : EG_a
 ego group of node b : EG_b
 social group of node a : SG_a
 social group of node b : SG_b
 threshold parameters: T_{eg}, T_{sg}

output: SG_a

when node a encounters node b

$$(1) \ Set_1 \leftarrow EG_a \cap EG_b$$

$$(2) \ P_{eg} \leftarrow \min \left\{ \frac{|Set_1|}{|EG_a|}, \frac{|Set_1|}{|EG_b|} \right\}$$

$$(3) \ Set_2 \leftarrow EG_a \cap SG_b$$

$$(4) \ Set_3 \leftarrow SG_a \cap EG_b$$

$$(5) \ P_{sg} \leftarrow \min \left\{ \frac{|Set_2|}{|SG_b|}, \frac{|Set_3|}{|SG_a|} \right\}$$

(6) **If** a and b belong to same social group

(7) update SG_a with SG_b ;

(8) **Else if** $Friend_{(a,b)} > 0$ and $P_{eg} \geq T_{eg}$

(9) update SG_a with SG_b ;

(10) **Else if** $P_{sg} \geq T_{sg}$

(11) update SG_a with SG_b ;

(12) **End if**

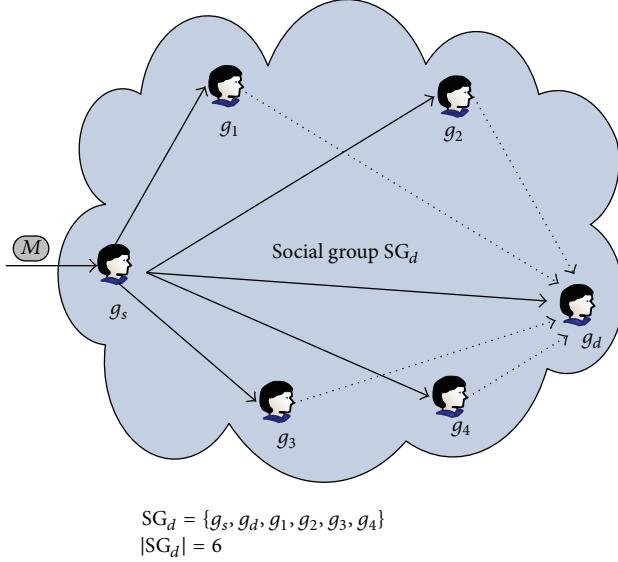
(13) **Return** SG_a

ALGORITHM 1: Social group establishment process on node a .

all social relationships, we further assume that each node belongs to and only belongs to one social group. The smallest social group contains only the node itself. In this case, adding a node to a social group is to merge the two social groups. As shown in line 9 and line 11, when adding node b to node a 's social group, the update process needs to merge the two social groups SG_a and SG_b . But in this moment, only the two nodes can capture and update the group changes. The other member nodes do know nothing about these changes. Fortunately, the member nodes of a social group are closely associated, and they encounter each other very frequently. We can make full use of these frequent encounter opportunities to capture and update those group changes. The update process of lines 6-7 is to finish this operation when encountering another member node of the same social group. Note that the case that a node moves out of its social group may also occur. If and only if a node is no longer a friend of any other social group member node, the node should be removed from the social group. This kind of information should also be captured and updated by the update process (i.e., the update process in lines 6-7) in every encounter chance.

3.4. Social Group Based Flooding Model. Based on the above social group information, we can finally implement our flooding routing model. Here we first define the node sets N_i and SG_d , where N_i denotes the node set that have been encountered by node i , and SG_d represents the target social group where destination node d resides.

The core strategy in our routing model is that message is flooded to all member nodes once the message is propagated to the target social group where the destination node resides,

FIGURE 2: A two-virtual-hop flooding model of SG_d .

which we call social group based flooding. For the purpose of successfully and quickly spreading message into the target social group, we also need to seek help from some intermediate relay nodes. Consequently, the choice of these intermediate relay nodes will greatly affect final routing performance and becomes the emergent thing we need to tackle with. In order to select fewer but better intermediate relay nodes, we attempt to compute the potential likelihood that a node can finish message transmission based on social group flooding. For this purpose, we first need to model group flooding process as a two-virtual-hop routing model. Then we can easily compute the potential likelihood of successful message transmission when the message has been delivered into the target social group through a group member (e.g., node g_s in Figure 2). Figure 2 shows an example of the two-virtual-hop model of the target social group SG_d and describes the possible group flooding process starting from node g_s . Concretely, a solid arrow in the figure represents a direct end-to-end path from g_s to an intermediate member node and a dashed arrow represents a virtual link from the intermediate member node to destination node g_d . Each virtual link consists of all the possible routes from the intermediate member node to g_d that do not pass through g_s . In other words, all these possible routes that do not pass through node g_s are abstracted as a virtual link.

In the above abstracted model, there will be $|SG_d| - 1$ different virtual routes from g_s to destination node g_d once message is delivered into the target social group SG_d through the member node g_s . For the path $g_s \rightarrow g_i \rightarrow g_d$, we use $P_{g_s \rightarrow g_i}(g_d)$ to represent the delivery possibility that g_s can successfully deliver message to destination g_d . Likewise, $P_{g_i^-}(g_i, g_d)$ is used to represent the total possibility that node g_i can deliver message to g_d along the virtual link $g_i \rightarrow g_d$

in the case of not passing through g_s . Obviously, we have the following equation:

$$P_{g_s \rightarrow g_i}(g_d) = P_{(g_s, g_i)} \times P_{g_i^-}(g_i, g_d). \quad (5)$$

And for the direct end-to-end path $g_s \rightarrow g_d$, we also have

$$\begin{aligned} P_{g_i^-}(g_d, g_d) &= 1.0 \\ P_{g_s \rightarrow g_d}(g_d) &= P_{(g_s, g_d)} \times P_{g_i^-}(g_d, g_d) = P_{(g_s, g_d)}. \end{aligned} \quad (6)$$

The message may be successfully delivered through any one of the above $|SG_d| - 1$ virtual paths. So taking into account all of them, we can compute the total delivery possibility $P_{(g_s, g_d)}^{SG_d}$ that g_s can successfully deliver a message to destination node g_d by flooding the message among the target social group SG_d :

$$\begin{aligned} P_{(g_s, g_d)}^{SG_d} &= 1 - \prod_{\substack{g_i \in SG_d \\ g_i \neq g_s}} (1 - P_{g_s \rightarrow g_i}(g_d)) \\ &= 1 - \prod_{\substack{g_i \in SG_d \\ g_i \neq g_s}} (1 - P_{(g_s, g_i)} \times P_{g_i^-}(g_i, g_d)). \end{aligned} \quad (7)$$

Here, if $g_s = g_d$, we set

$$P_{(g_s, g_s)}^{SG_d} = P_{(g_d, g_d)}^{SG_d} = 1.0. \quad (8)$$

Now, in order to get the final value of $P_{(g_s, g_d)}^{SG_d}$, we only need to compute $P_{g_i^-}(g_i, g_d)$. Since the virtual link from g_i to g_d does not pass through node g_s , the longest path will comprise $|SG_d| - 1$ nodes and the shortest route only comprises 2 nodes. Consequently, the number of all possible paths from g_i to g_d in the virtual link is C_A :

$$\begin{aligned} C_A &= A^0_{|SG_d|-3} + A^1_{|SG_d|-3} + \cdots + A^{|SG_d|-4}_{|SG_d|-3} + A^{|SG_d|-3}_{|SG_d|-3} \\ &= \sum_{i=0}^{|SG_d|-3} A^i_{|SG_d|-3}. \end{aligned} \quad (9)$$

But due to the strictly limited processing capability in DTNs, it is difficult to compute all delivery possibilities of these routes. Besides, for most DTN applications, the network bandwidth and buffer resource are very precious. Consequently, one optimization goal of DTN routing is to control average hop count for the purpose of further reducing the cost of message transmission. Under this circumstance, we further simplify the virtual link by controlling the maximum hop count. In other words, we only take into account the delivery likelihood within two hops scope when computing $P_{g_i^-}(g_i, g_d)$. If the path from g_i to g_d is only one hop, we get the delivery probability

$$P_{g_i^-}^{1_{\text{hop}}}(g_i, g_d) = P_{(g_i, g_d)}. \quad (10)$$

If the path from g_i to g_d is only two hops, we get the delivery probability

$$P_{g_s^-}^{2\text{hop}}(g_i, g_d) = 1 - \prod_{\substack{g_j \in SG_d \\ g_j \neq g_s \\ g_j \neq g_i \\ g_j \neq g_d}} (1 - P_{(g_i, g_j)} \times P_{(g_j, g_d)}). \quad (11)$$

Now within two hops scope, we can finally get

$$\begin{aligned} P_{g_s^-}(g_i, g_d) &= 1 - \left(1 - P_{g_s^-}^{1\text{hop}}(g_i, g_d)\right) \times \left(1 - P_{g_s^-}^{2\text{hop}}(g_i, g_d)\right) \\ &= 1 - \left(1 - P_{(g_i, g_d)}\right) \prod_{\substack{g_j \in SG_d \\ g_j \neq g_s \\ g_j \neq g_i \\ g_j \neq g_d}} (1 - P_{(g_i, g_j)} \times P_{(g_j, g_d)}). \end{aligned} \quad (12)$$

Here, we analyze the cost of computing $P_{g_s^-}(g_i, g_d)$. When computing $P_{g_s^-}^{1\text{hop}}(g_i, g_d)$, we can easily get the cost

$$\text{Cost}_{1\text{hop}} = O(1). \quad (13)$$

To compute $P_{g_s^-}^{2\text{hop}}(g_i, g_d)$, we need to do $(|SG_d| - 3) + (|SG_d| - 4)$ multiplication operations and $|SG_d| - 2$ subtraction operations. So we get

$$\begin{aligned} \text{Cost}_{2\text{hop}} &= (|SG_d| - 3) + (|SG_d| - 4) + (|SG_d| - 2) \\ &= 3|SG_d| - 9 = O(|SG_d|). \end{aligned} \quad (14)$$

Then we can get the total cost when computing $P_{g_s^-}(g_i, g_d)$:

$$\text{Cost} = \text{Cost}_{1\text{hop}} + \text{Cost}_{2\text{hop}} + 4 = O(|SG_d|). \quad (15)$$

By controlling the cost to $O(|SG_d|)$ level, DTN node can easily compute $P_{g_s^-}(g_i, g_d)$. And then with (7), node can finally compute the delivery likelihood $P_{(g_s, g_d)}^{SG_d}$. For this purpose, every social group should maintain a probability matrix PM_{SG_d} (as described below) to record and update the encounter possibility between two group member nodes. And finally we can get the matrix PM_{SG_d, SG_d} , in which the element in i th row and j th column record the $P_{(g_i, g_j)}^{SG_d}$ for nodes g_i and g_j :

$$\begin{aligned} \text{PM}_{SG_d} &= \begin{bmatrix} P_{(g_1, g_1)} & \cdots & P_{(g_1, g_{|SG_d|})} \\ \vdots & \ddots & \vdots \\ P_{(g_{|SG_d|}, g_1)} & \cdots & P_{(g_{|SG_d|}, g_{|SG_d|})} \end{bmatrix} \\ P_{(g_i, g_i)} &= 1.0, \quad 1 \leq i \leq |SG_d| \\ \text{PM}_{SG_d, SG_d} &= \begin{bmatrix} P_{(g_1, g_1)}^{SG_d} & \cdots & P_{(g_1, g_{|SG_d|})}^{SG_d} \\ \vdots & \ddots & \vdots \\ P_{(g_{|SG_d|}, g_1)}^{SG_d} & \cdots & P_{(g_{|SG_d|}, g_{|SG_d|})}^{SG_d} \end{bmatrix} \\ P_{(g_j, g_j)}^{SG_d} &= 1.0, \quad 1 \leq j \leq |SG_d|. \end{aligned} \quad (16)$$

Now with PM_{SG_d, SG_d} at hand, we can predict the potential delivery likelihood that an intermediate node (does not belong to the target social group SG_d) can successfully finish message transmission in the assistance of the target social group members it encounters. Afterwards, in order to successfully and quickly spread message into the target social group, current node will select an intermediate node as the next hop relay node if the node has a bigger delivery likelihood than current node. Now we assume that current node i has a message destined for node g_d and node i does not belong to the target social group SG_d . When encountering a member node of SG_d , node i will certainly deliver the message to the member node. Then the message will be flooded among the target group SG_d until finishing message transmission. But when meeting an intermediate node, which does not belong to the target social group, current node needs to decide whether to deliver the message to the encountered node. For the convenience of describing the detailed process, we define the node set $NG_{i,d}$ as follows, which represents the group member nodes of SG_d encountered by node i :

$$\begin{aligned} NG_{i,d} &= N_i \cap SG_d \\ &= \{j \mid P_{(i,j)} > 0, \quad j \in SG_d\}. \end{aligned} \quad (17)$$

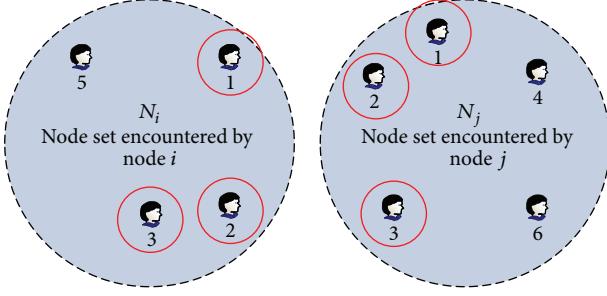
Here we can get the matrix M_I , which records the encounter possibilities between node i and all the nodes in $NG_{i,d}$. And then we get the matrix M_{SG_d} from PM_{SG_d, SG_d} , which records the $P_{(g_j, g_d)}^{SG_d}$ for each $g_j \in NG_{i,d}$ and destination node g_d . After transforming M_I to the diagonal matrix M_I^{diag} , we can compute the final matrix M_P by the matrix of $M_{N_i}^{\text{diag}} \times M_{SG_d}$:

$$M_I = \begin{bmatrix} P_{(i, g_1)} & \cdots & P_{(i, g_j)} & \cdots & P_{(i, g_{|NG_{i,d}|})} \end{bmatrix}$$

$$M_{SG_d} = \begin{bmatrix} P_{(g_1, g_d)}^{SG_d} \\ \vdots \\ P_{(g_j, g_d)}^{SG_d} \\ \vdots \\ P_{(g_{|NG_{i,d}|}, g_d)}^{SG_d} \end{bmatrix},$$

$$M_I^{\text{diag}} = \text{diag}(M_I)$$

$$= \begin{bmatrix} P_{(i, g_1)} & & & \\ & \ddots & & \\ & & P_{(i, g_j)} & \\ & & & \ddots \\ & & & & P_{(i, g_{|NG_{i,d}|})} \end{bmatrix},$$

FIGURE 3: Encounter history of node i and node j .

$$\begin{aligned}
 M_P &= M_I^{\text{diag}} \times M_{\text{SG}_d} \\
 &= \begin{bmatrix} P_{(i,g_1)} \times P_{(g_1,g_d)}^{\text{SG}_d} \\ \vdots \\ P_{(i,g_j)} \times P_{(g_j,g_d)}^{\text{SG}_d} \\ \vdots \\ P_{(i,g_{|\text{NG}_{i,d}|})} \times P_{(g_{|\text{NG}_{i,d}|},g_d)}^{\text{SG}_d} \end{bmatrix} \quad (18)
 \end{aligned}$$

Now, each element in M_P denotes the possibility that node i can successfully deliver message to destination g_d through a certain target social group member in $\text{NG}_{i,d}$. Finally with the matrix M_P , we can get the potential possibility $P_{(i,g_d)}^L$ that intermediate node i (does not belong to the target group) can successfully finish message transmission in the assistance of the target social group members it will encounter in the near future. Obviously, we have

$$P_{(i,g_d)}^L = 1 - \prod_{g_j \in \text{NG}_{i,d}} \left(1 - P_{(i,g_j)} \times P_{(g_j,g_d)}^{\text{SG}_d} \right). \quad (19)$$

3.5. Message Redundancy Control Model. In most DTN application scenarios, the node buffer resource is extremely limited, which determines that the number of message copies should be controlled to get a better routing performance. As a result, improving the utilization efficiency of network resources for each message copy can further increase message delivery ratio. For this purpose, we make attempts to reduce message redundancy when selecting intermediate relay nodes to spread message to target social group. Assuming that node i and node j both carry a same message copy, then for the node encountered by both i and j , one of the two message copies is redundant. Concretely, if most nodes that node i may meet will also be encountered by node j , then one of the two message copies is considered to be unnecessary. Here based on these considerations, we propose a message redundancy model to avoid creating such redundant message copies when selecting next hop relay nodes.

Here, we assume that the nodes encountered by a node in the past are possible to be encountered again. Then we use the encountered node set to predict the node set that a node will encounter in the near future. For current node i and its

neighbor node j , we denote the node set $N_i \cap N_j$ as the nodes that will be encountered by both node i and node j , which is node set $\{1, 2, 3\}$ as shown in Figure 3:

$$N_i \cap N_j = \{k \mid P_{(i,k)} > 0, P_{(j,k)} > 0\}. \quad (20)$$

Now, we describe the detailed message redundancy model as follows. For a node k , there are two cases to occur: k will be encountered by both i and j in the near future or not. We represent the two cases using

$$y_k = \begin{cases} 1 & \text{if } i \text{ and } j \text{ will both encounter } k \\ 0 & \text{else.} \end{cases} \quad (21)$$

According to encounter probabilities between nodes, we can predict the probability that i and j both encounter k . Then through a subtraction operation, we can get the probability of the other case. Obviously, we have

$$\begin{aligned}
 P(y_k = 1 \mid N_i, N_j) &= P_{(i,k)} \times P_{(j,k)}, \\
 P(y_k = 0 \mid N_i, N_j) &= 1 - P(y_k = 1 \mid N_i, N_j). \quad (22)
 \end{aligned}$$

Assuming that node i replicates a message to node j , then we can predict whether there will be redundant message copies via node k by comparing the probabilities computed by (22). As defined below, if the probability of $y_k = 1$ is bigger than the probability of $y_k = 0$, then there will be a redundant message copy via node k (i.e., $Y_k = 1$):

$$Y_k = \begin{cases} 1 & \text{if } P(y_k = 1 \mid N_i, N_j) > P(y_k = 0 \mid N_i, N_j) \\ 0 & \text{if } P(y_k = 1 \mid N_i, N_j) \leq P(y_k = 0 \mid N_i, N_j). \end{cases} \quad (23)$$

Finally for each node k in $N_i \cap N_j$, we get the vector

$$Y = \langle Y_1 \cdots Y_k \cdots Y_{|N_i \cap N_j|} \rangle. \quad (24)$$

With the vector Y , we use (25) to compute the message redundancy ratio $\text{MR}_{(i,j)}$ for node j :

$$\text{MR}_{(i,j)} = \frac{1}{|N_j|} \sum_{k \in N_i \cap N_j} Y_k. \quad (25)$$

It indicates that node i has been able to cover almost all nodes that node j will encounter once the value of $\text{MR}_{(i,j)}$ is close to 1. In this case, node j should not be selected as next hop relay node for avoiding message redundancy. On the contrary, node j is considered to be a good choice if the value of $\text{MR}_{(i,j)}$ is close to 0 since node i and node j have different social circles in that case. To address this problem, we define a message redundancy variable named as $\text{MR}_{\text{threshold}} \in (0, 1)$, which can be flexibly set to an appropriate value according to specific application scenario. When $\text{MR}_{(i,j)} < \text{MR}_{\text{threshold}}$, node j is considered as a good choice to act as a relay node and vice versa. Aiming to further control network overhead ratio, we propose to set $\text{MR}_{\text{threshold}}$ to a small value.

```

Input:
one-hop neighbors of  $N_i$ : neigh_list
messages stored in  $N_i$ : msg_list
the value of  $MR_{threshold}$ : MR
output:
(1) For message in msg_list
(2)    $dest \leftarrow$  message.destination
(3)    $SG_d \leftarrow$  the target social group of  $dest$ 
(4)   For node in neigh_list
(5)      $EG \leftarrow$  the ego group of node
(6)     If node =  $dest$  OR node  $\in SG_d$  OR  $dest \in EG$ 
(7)       replicate message to node
(8)     Else if  $N_i \in SG_d$ 
(9)       If  $P_{(node, dest)}^L > P_{(N_i, dest)}^{SG_d}$  AND  $MR_{(N_i, node)} < MR$ 
(10)        replicate message to node
(11)      End if
(12)    Else
(13)      If  $P_{(node, dest)}^L > P_{(N_i, dest)}^L$  AND  $MR_{(N_i, node)} < MR$ 
(14)        replicate message to node
(15)      End if
(16)    End if
(17)  End for
(18) End for

```

ALGORITHM 2: DGA routing algorithm on node N_i .

3.6. Detailed DGA Routing Algorithm. Based on the above works, we finally implement our routing algorithm as illustrated in Algorithm 2. There are two cases that current node should deliver message to a node: one is that the node belongs to the social group of destination node, the other is that destination node belongs to the node's ego group. Lines 6–7 are to test these cases and finish message transmission. Otherwise, by using the proposed social group based flooding model to compute the potential delivery probability and using the message redundancy control model to reduce redundant messages, lines 8–14 select fewer but better relay nodes to spread message copies in order to quickly deliver message to the target social group.

4. Simulation

By using the ONE [17] simulator, we conduct extensive simulations to evaluate the performance of DGA under various settings. The compared algorithms, simulation settings, evaluation metrics, and results are described as follows.

4.1. Compared Algorithms. For evaluations in this paper, we only focus on the routing algorithms similar to DGA. Firstly, considering that DGA takes advantage of social behaviors among nodes, we add the typical social-based routing algorithm Bubble Rap [12] to the comparisons. This can evaluate the improvements of DGA in terms of using social properties. Secondly, taking into account that DGA abstracts all social relationships and further uses encounter probability to measure relationship strength, we also add ProPHET to the evaluations. The concept of encounter

probability presented by ProPHET can be considered the most essential form of social relationships. So in this case, ProPHET can also be regarded as a social tie based routing algorithm. The concept of abstracting social relationships is also from ProPHET. Lastly, the core routing strategy of DGA is flooding based on social group, which comes in parts from Epidemic but different from Epidemic. Introducing Epidemic to the simulations is to find out the improvements of local flooding (DGA) compared to typical global flooding (Epidemic).

4.2. Simulation Settings. To evaluate the routing performance of DGA based on social behaviors, we conduct all simulations on synthetic traces generated by the working day movement (WDM) model [18]. This is because the WDM model brings more reality to the human movement by modeling three major activities typically performed by humans during a working week: sleeping at home, working at the office, and going out with friends. Beyond the activities themselves, the WDM model also includes different transport models. The nodes can move alone or in groups by walking, riding, or driving. The ability to move alone or in groups at different speeds increases the heterogeneity of movement which has impact on the performance of, for example, routing protocols. In addition, WDM introduces communities and social relationships. The communities are composed from nodes which work in the same office, spend time in the same evening activity spots, or live together. Moreover, we can modify the model parameters as needed, so that it can reproduce various empirical mobility properties, which is beneficial to

TABLE 1: Default simulation settings.

Parameter	Default value
Area size	10000 m × 8000 m
Message size	500 K–1 M
Message interval	30–40 s
Node buffer size	80 M
Time-to-live (TTL)	12 hours
Simulation time	5 days
P_{friend}	0.95
T_{eg}	0.5
T_{sg}	0.5
MR _{threshold}	0.5
P_{init}	0.75
γ	0.98
β	0.25

the routing performance evaluations. This is another reason that we use the traces generated by WDM.

In the trace-driven simulations, there are 50 working offices and 10 meeting spots. Working day length is 4 hours and office wait time is about 10 minutes to 4 hours. The probability of shopping after work is 0.5 and shopping time is about 1–2 h. We totally use 110 nodes, which consist of 10 bus nodes and 100 human nodes. In detail, bus nodes are based on bus movement model and divided into 5 bus groups. Each bus group travels along different traffic routes and consists of two bus nodes. Each bus node moves with speed of 7–10 m/s, wait time of 10–30 s, transmit speed of 10 Mbps, and transmit range of 1000 m. The human nodes are based on WDM model and also divided into 5 groups. Each human group consists of 20 nodes and they have their own meeting spots, homes, and working offices. Each human node moves with speed of 0.8–1.4 m/s, transmit speed of 2 Mbps, and transmit range 20 m. All other default simulation settings are shown in Table 1.

The simulations are grouped into the three categories: varying buffer size, varying message's time-to-live, and varying message generation internal. When varying a single setting parameter in simulations, the other setting parameters follow the default settings in Table 1. Firstly, we investigated the routing performance of the four algorithms when varying buffer size from 20 M to 120 M. Secondly, we evaluated their routing performance when varying message's time-to-live from 4 hours to 24 hours. Lastly, we conducted the simulations when varying the message generation interval from 20 seconds to 100 seconds. All evaluation results are shown in Section 4.4.

4.3. Evaluation Metrics. Under the same guideline, we evaluated the above four routing algorithms based on the following metrics.

- (1) Delivery ratio: normally, the ultimate routing goal in DTNs is to successfully deliver message to its destination. This metric is the measure of delivery capability for each algorithm.

- (2) Overhead ratio: it is desirable to have a low overhead ratio, since it reflects the efficiency of message transmission.
- (3) Average latency: end-to-end latency is another important routing goal. A lower average latency means a better routing performance.
- (4) Average hop count: minimizing the number of hops that a message must take in order to reach the destination is a routing goal to reduce transmission cost, such as bandwidth and energy.

4.4. Simulation Results

4.4.1. Performance Evaluations by Varying Buffer Size. Regarding the simulation results in Figure 4, DGA achieves the highest delivery ratio and the lowest overhead ratio. The average hop count of DGA is almost as few as Bubble Rap and is fewer than Epidemic and ProPHET. In addition, DGA is also able to get some advantages in delivery latency compared to Epidemic and ProPHET. All these results can verify the great improvements of DGA in routing performance.

As shown in Figures 4(a) and 4(b), DGA can outperform Bubble Rap, Epidemic, and ProPHET in terms of message delivery ratio and overhead ratio. This is because DGA takes full use of the gregariousness characteristic of moving nodes. Based on the proposed ego group model and social group model, DGA can find out the friend nodes that are closely associated with destination node and then only floods message among the target social group. Besides, based on the social group based flooding model, current node is also able to select better nodes as next hop relay nodes to quickly and successfully deliver message to the target social group. In this case, DGA achieves a higher delivery ratio than Bubble Rap. In addition, the proposed message redundancy control model further helps DGA to get a lower overhead ratio. DGA and ProPHET both use the encounter probability to measure the strength of the social tie between two nodes, but ProPHET does not take further optimization strategy to improve routing performance. On the contrary, ProPHET just naively delivers message to the encountered nodes that have bigger probabilities than current node. Therefore the routing performance of ProPHET is worse than DGA. Compared to the global flooding strategy of Epidemic, DGA accurately controls the flooding scope based on social group model, thus greatly avoiding creating too many redundant messages and further controlling network overhead ratio. And DGA takes into consideration nodes' social relationships for improving routing performance. As a result, DGAs significantly outperform Epidemic.

From Figures 4(c) and 4(d) we can see that the average latency and average hop count of DGA and Bubble Rap are lower than those of Epidemic and ProPHET when buffer size is more than 40 M, which reflects the fact that the routing strategies of DGA and Bubble Rap are more accurate, thus efficiently reducing the cost of message transmission.

Finally from the whole Figure 4, we can make a conclusion that DGA can outperform Epidemic and ProPHET in social networks in terms of delivery ratio, overhead ratio,

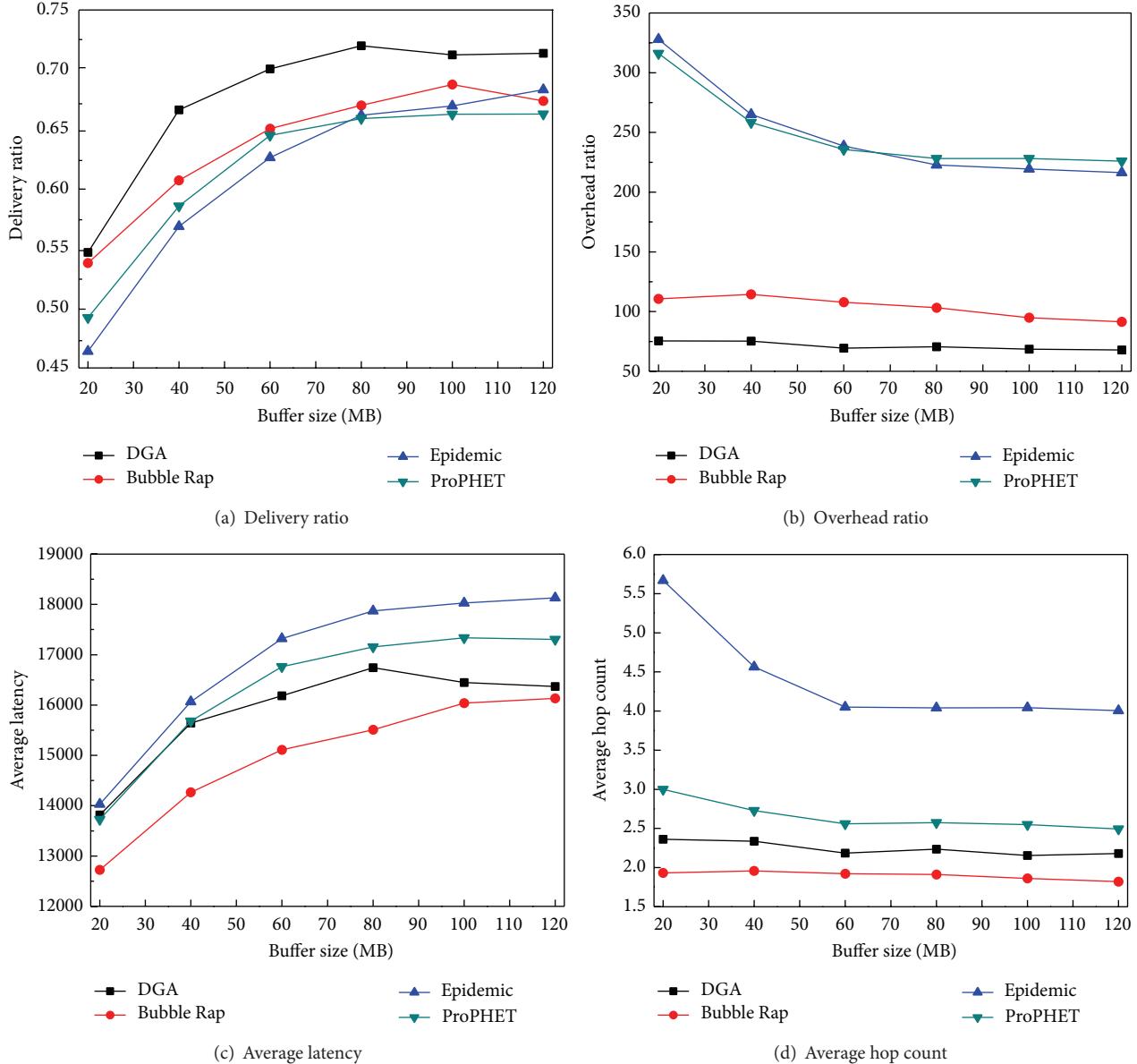


FIGURE 4: Delivery ratio, overhead ratio, average latency, and average hop count versus buffer size (WDM).

average latency, and average hop count. And although the average latency and average hop count of DGA are slightly higher than those of Bubble Rap, the message delivery ratio of DGA is higher than Bubble Rap and the overhead ratio of DGA is only about 50% of that of Bubble Rap. As a result, from the point of view of high delivery ratio and low overhead ratio, DGA is able to outperform Bubble Rap.

4.4.2. Performance Evaluations by Varying Message TTL. The results in Figure 5 show that DGA gets the highest delivery ratio when message time-to-live is more than 8 hours and achieves the lowest overhead ratio when message TTL is less than 16 hours. As same as shown in Figure 4, DGA can still achieve satisfying delivery latency and average hop count.

These simulation results prove once again the improvements and routing efficiency of DGA.

From Figure 5(a), we can find that message TTL can significantly affect the message delivery ratios of the four algorithms when buffer size is relatively sufficient. When message TTL is less than 12 hours, the delivery ratios of the four algorithms all keep increasing. This is because that message has a longer time to seek destination node in this case. But when message TTL is more than 12 hours, their delivery ratios keep decreasing. The reason is that long message TTL also leads to the presence of more messages, which exacerbates the consumption of cache resource. However, DGA can keep a higher delivery ratio compared to Bubble Rap, Epidemic, and ProPHET when message TTL is more than 8 hours. To a certain degree, Bubble Rap is able to avoid

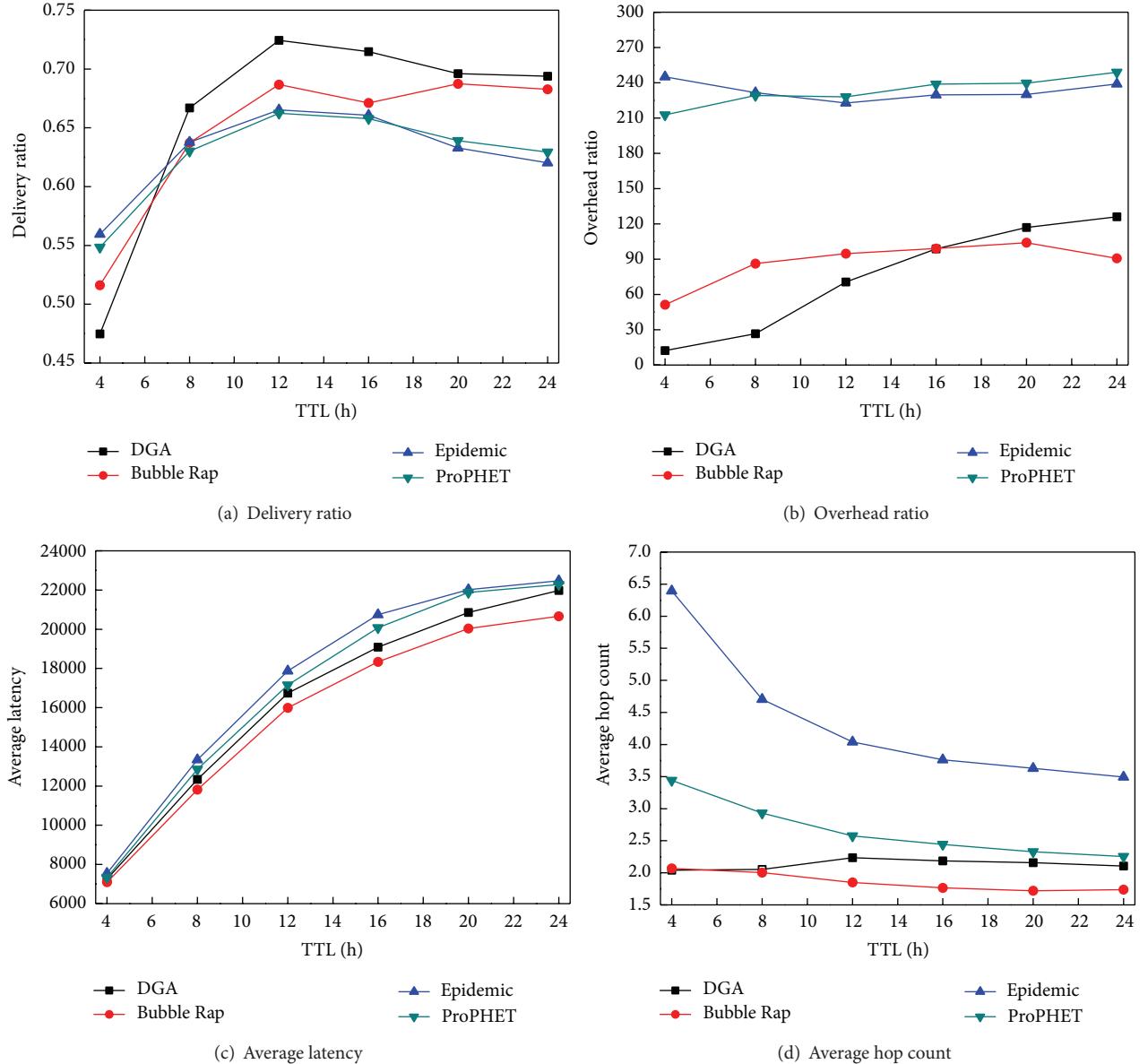


FIGURE 5: Delivery ratio, overhead ratio, average latency, and average hop count versus message TTL (WDM).

generating too many redundant messages by selecting relay nodes based on social behaviors of nodes. But DGA can further reduce redundant messages by using the proposed message redundancy model while selecting relay nodes based on social behaviors. On the contrary, Epidemic and ProPHET do not take efficient strategy to control message redundancy. In this case, they will inevitably create many redundant messages. Therefore from the perspective of the message redundancy, the increases of message TTL help DGA to get a higher delivery ratio. In Figures 5(b), 5(c) and 5(d), DGA and Bubble Rap can also keep their advantages in overhead ratio, delivery latency, and average hop count with the increases of message TTL.

To sum up, considered from message delivery ratio and network overhead ratio, DGA can outperform Epidemic

and ProPHET and achieve certain advantages compared to Bubble Rap.

4.4.3. Performance Evaluations by Varying Message Generation Interval. As shown in Figure 6, DGA achieves the highest message delivery ratio and the lowest overhead ratio when message generation interval is more than 20 seconds. In addition, DGA gets significant advantages in delivery latency and average hop count compared to Epidemic and ProPHET. From these results, we can verify the great improvements of DGA.

In Figure 6(a), the delivery ratios of the four routing algorithms all keep increasing with the increase of message generation interval, but DGA can keep the highest delivery

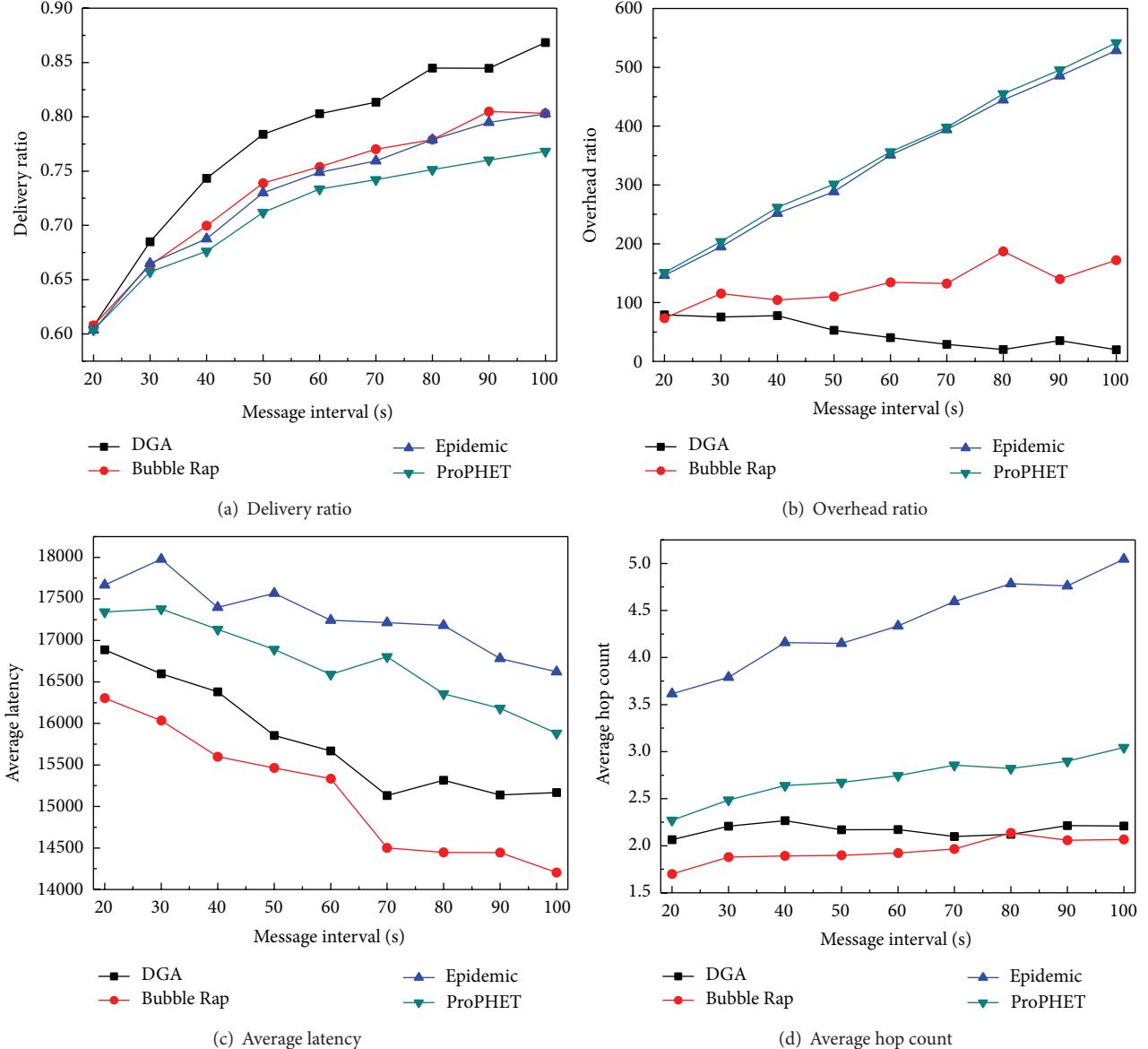


FIGURE 6: Delivery ratio, overhead ratio, average latency, and average hop count versus message interval (WDM).

ratio. The key reason is that the number of new messages generated by network will be reduced when message generation interval keeps increasing, thus weakening the competition for cache resource. In this case, all algorithms can use these relatively sufficient buffer resources to better deploy their routing strategies. Furthermore, DGA can efficiently select fewer but better relay nodes to deliver message to the target social group based on the proposed social group flooding model and find out the nodes that are closely associated with destination node for locally flooding message based on social group model. Thus, DGA can get a higher delivery ratio. Moreover, in Figure 6(b), the overhead ratio of DGA keeps decreasing, while the overhead ratios of the other three algorithms are increasing. This is because DGA can further control message redundancy by using the proposed message redundancy control model, which can verify the efficient

improvements of DGA in controlling network overhead ratio. Consequently, DGA gets a lower overhead ratio with the increase of message generation interval.

Similar to Figures 4 and 5, DGA can get the lower delivery latency and fewer average hop count compared to Epidemic and ProPHET, although the delivery latency and average hop count of DGA are still slightly higher than those of Bubble Rap. But taking into account the message delivery ratio and overhead ratio, DGAs still outperform the other three routing algorithms.

5. Conclusion

In this paper, we proposed the dynamic groups based adaptive DTN routing algorithm DGA for social networks. Taking

into consideration a variety of social relations and their different forms, we first abstract these social relationships and then measure them using the encounter probability. In order to take full use of the gregariousness characteristic of moving nodes, we define the ego group model and social group model to dynamically establish different social groups. Based on social group information, we only flood message among the target social group where destination node resides, thus efficiently controlling the flooding scope in the case of improving message delivery ratio. In order to quickly and successfully deliver message to its target social group, we proposed a social group based flooding model to help current node to select better next hop relay nodes. Furthermore, we also present a message redundancy control model to avoid generating unnecessary message copies, thus further controlling message redundancy and reducing network overhead ratio. Extensive simulation results show that the proposed DGA can outperform Bubble Rap, Epidemic, and ProPHET in terms of message delivery ratio and overhead ratio.

Conflict of Interests

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

Acknowledgments

This research is supported in part by Foundation Research Project of Qingdao Science and Technology Plan under Grant no. 12-1-4-2-(14)-jch and Natural Science Foundation of Shandong Province under Grant no. ZR2013FQ022.

References

- [1] V. N. G. J. Soares, J. J. P. C. Rodrigues, and F. Farahmand, "GeoSpray: a geographic routing protocol for vehicular delay-tolerant networks," *Information Fusion*, vol. 15, pp. 102–113, 2014.
- [2] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *Proceedings of the ACM SIGCOMM Workshops: Conference on Computer Communications*, pp. 244–251, August 2005.
- [3] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, "Diversity of forwarding paths in pocket switched networks," in *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference (IMC '07)*, pp. 161–174, October 2007.
- [4] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *SIGMOBILE Mobile Computing Communications Review*, vol. 11, no. 4, pp. 23–33, 2007.
- [5] V. Nikolaos and Y. Kun, "Mobile social networks: architecture, social properties, and key research challenges," *IEE Communications Surveys & Tutorials*, vol. 15, no. 3, 2013.
- [6] N. Kayastha, D. Niyato, P. Wang, and E. Hossain, "Applications, architectures, and protocol design issues for mobile social networks: a survey," *Proceedings of the IEEE*, vol. 99, no. 12, pp. 2130–2158, 2011.
- [7] X.-M. Fan, Z.-G. Shan, B.-X. Zhang, and H. Chen, "State-of-the-art of the architecture and techniques for delay-tolerant networks," *Acta Electronica Sinica*, vol. 36, no. 1, pp. 161–170, 2008.
- [8] K. Fall and S. Farrell, "DTN: an architectural retrospective," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 5, pp. 828–836, 2008.
- [9] V. Cerf, S. Burleigh, A. Hooke et al., *Delay-Tolerant Networking Architecture*, RFC4838, IETF, 2007.
- [10] A. Vahdat and D. Becker, *Epidemic Routing For Partially Connected Ad Hoc Networks*, Duke University, Durham, NC, USA, 2000.
- [11] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mobile Computing Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.
- [12] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: social-based forwarding in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576–1589, 2011.
- [13] J. Wu and Y. Wang, "Hypercube-based multi-path social feature routing in human contact networks," *IEEE Transactions on Computers*, vol. 63, no. 2, pp. 383–396, 2014.
- [14] F. Xia, A. Ahmed, L. Yang, J. Ma, and J. Rodrigues, "Exploiting social relationship to enable efficient replica allocation in Ad-hoc social networks," *IEEE Transactions on Parallel and Distributed Systems*, 2014.
- [15] M. J. Xiao, J. Wu, and L. S. Huang, "Community-aware opportunistic routing in mobile social networks," *IEEE Transactions on Computers*, 2013.
- [16] J. Wu, M. J. Xiao, and L. S. Huang, "Homing spread: community home-based multi-copy routing in mobile social networks," in *Proceedings of IEEE INFOCOM*, pp. 2319–2327, 2013.
- [17] A. Keranen, J. Ott, and T. Karkkainen, "The ONE simulator for DTN protocol evaluation," in *Proceeding of the Second International Conference on Simulation Tools and Techniques*, pp. 1–10, 2009.
- [18] F. Ekman, A. Keränen, J. Karvo, and J. Ott, "Working day movement model," in *Proceeding of 1st ACM/SIGMOBILE Workshop on Mobility Models for Networking Research*, pp. 33–40, 2008.

Research Article

A Flow-Partitioned Unequal Clustering Routing Algorithm for Wireless Sensor Networks

Jian Peng,¹ Xiaohai Chen,¹ and Tang Liu^{1,2}

¹ College of Computer Science, Sichuan University, Chengdu 610065, China

² College of Fundamental Education, Sichuan Normal University, Chengdu 610068, China

Correspondence should be addressed to Tang Liu; crikey@163.com

Received 6 March 2014; Accepted 13 March 2014; Published 17 April 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Jian Peng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Energy efficiency and energy balance are two important issues for wireless sensor networks. In previous clustering routing algorithms, multihop transmission, sleep scheduling, and unequal clustering are always used to improve energy efficiency and energy balance. In these algorithms, only the cluster heads share the burden of data forwarding in each round. In this paper, we propose a flow-partitioned unequal clustering routing (FPUC) algorithm to achieve better energy efficiency and energy balance. FPUC consists of two phases: clustering and routing. In the clustering phase, the competition radius is computed according to the node density and the distance from sensor nodes to the sink. The sensor nodes that have more residual energy and larger overlapping degree have higher probability to be selected as cluster heads. In the routing phase, each cluster head first finds the gateway nodes and then distributes the data flow to each of its gateway nodes depending on residual energy. After that, each gateway node forwards the data to the next hop with minimum cost. Two metrics called network lifetime and coverage lifetime are used to compare the performance of FPUC with that of the existing ones. Simulation results show that FPUC can achieve longer network lifetime and coverage lifetime than previous algorithms.

1. Introduction

With recent advancement in techniques such as microelectromechanical, embedded, and low-power design, wireless sensor networks (WSNs) have achieved great developments, which have been widely used in areas such as environmental monitoring, health care, and battlefield surveillance [1]. WSNs are collections of some sensor nodes that have integrated wireless communication, sensing, data storage, and data processing capabilities. These sensor nodes transmit sensed data to the sink via single-hop or multihop routing to accomplish specific tasks. Generally speaking, each sensor node has limited energy because it is often powered by nonrechargeable batteries. So with time elapsing, some sensor nodes may run out of energy, thus leading to the phenomenon that other sensor nodes cannot send data to the sink, called *energy hole* [2–4]. Therefore, designing energy-efficient routing algorithm to prolong the lifetime is very important to wireless sensor networks.

Researchers have done a lot of related work, among which clustering, multihop transmission, and sleep scheduling are the most widely used techniques to improve energy efficiency. In clustering algorithms, each cluster consists of one cluster head (CH) and some cluster members (CM). CH aggregates the data received from CMs and then conveys the aggregated data to the sink. For the reason that data aggregation is applied, the amount of the whole network's data to be transmitted is significantly reduced, thus prolonging the lifetime [5–7].

Studies in [6–11] have shown that multihop transmission can effectively help reduce the energy consumed in communication and extend the lifetime. However, CHs closer to the sink would consume more energy because they are loaded with more relaying traffic than the other CHs. Therefore, the energy hole problem will arise around the sink much easier. To solve the problem, unequal clustering is proposed. In unequal clustering algorithms, the clusters closer to the sink have smaller cluster size. Actually node density around

the CH will also have effects on its energy consumption, especially under the nonuniform distribution. As Figure 1 shows, areas *a* and *b* have approximately equal distance to the sink, but node density of area *a* is obviously larger than that of area *b* which will lead to those CHs in area *a* having heavier burden than CHs in area *b*, and thus CHs in area *a* will consume more energy and then the energy hole may occur.

In [12–15], researchers have proposed some algorithms which aim to conserve energy by turning off as many as possible sensor nodes while achieving the application's coverage goal at the same time. Tao et al. first propose the flow-balanced routing protocol (FBR) [16], which is very different from traditional routing algorithms. In FBR, a graph topology is constructed instead of tree structure and each CH may have multiple parents. When CH relays data, it computes the amount of flow from itself to each of its parent according to parent's current residual energy to balance relaying node's energy consumption. However, each cluster has the same cluster size in FBR; thus sensor nodes closer to the sink will die faster.

In this paper, we propose a flow-partitioned unequal clustering routing algorithm (FPUC) aiming to achieve longer *network lifetime* and *coverage lifetime*. Network lifetime is defined as the duration from the beginning to the time when any or a given percentage of sensor nodes die. Coverage lifetime measures the period from the network setup time to the time that network coverage drops below a predefined threshold or application's demanding value [16–18]. Compared with the existing clustering routing algorithms, FPUC mainly has the following differences.

- (1) FPUC considers not only sensor node's residual energy but also sensor node's overlapping degree in the process of cluster heads selection. The sensor node which has more residual energy and larger overlapping degree has higher probability to be a cluster head.
- (2) FPUC computes the competition radius according to sensor node's distance to the sink and sensor node's surrounding node density.
- (3) After aggregating the data received from CMs, CH sends data to its CMs by the way of partitioning flow. When a CM receives data from its CH, it will transmit the flow to the next hop with the minimum cost. Thus the CH and CMs bear the burden of data relaying together.

The remainder of this paper is organized as follows. Section 2 introduces the related work. The network model and some definitions are described in Section 3. In Section 4, we propose the FPUC algorithm. Section 5 provides performance analysis about FPUC. In Section 6, we present the simulation results to show the validity of FPUC. The last section gives the conclusions.

2. Related Work

A lot of clustering routing algorithms have been proposed in recent years. Low-energy adaptive clustering hierarchy

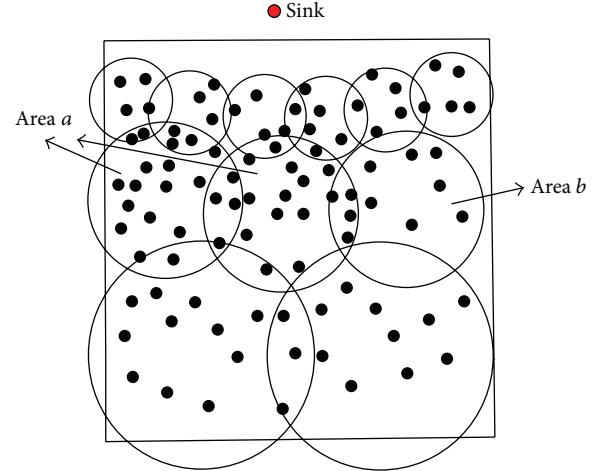


FIGURE 1: Clustering under nonuniform distribution.

(LEACH) [5] is the first clustering routing protocol proposed. In LEACH, each sensor node has a certain probability to become a cluster head per round and the CHs send the aggregated data to the sink directly. Obviously, LEACH is very simple and can effectively extend the lifetime. However, LEACH does not take the residual energy of sensor node into consideration which will result in that the sensor node with little residual energy may become a cluster head. Moreover, because of transmitting data to the sink via single hop, sensor nodes closer to the sink will die faster, respectively. To solve the problems of LEACH, Younis and Fahmy proposed HEED in which the sensor node having more residual energy has higher probability of becoming cluster head [6]. As a result, HEED can generate a more reasonable network topology. In HEED, it adopts multihop routing. Consequently, sensor nodes closer to the sink would deplete their energy faster. The unbalanced energy consumption among different CHs may cause the network to be partitioned.

For solving hot spots problems in multihop WSNs, Soro and Heinzelman first proposed unequal clustering algorithm [19]. In [19], the network topology is a two-level concentric circle around the sink and the inner ring has smaller cluster size than the outer ring. However, the work in the literature [19] focuses on a heterogeneous network where CHs are deployed at determined locations. In [7], Li et al. proposed an energy-efficient unequal clustering algorithm (EEUC). In EEUC, the cluster closer to the sink has smaller size so that the cluster head can have more energy to relay data. Other unequal clustering algorithms, for example, [20, 21], are, respectively, based on the core idea of EEUC. These existing algorithms compute competition radius only according to sensor node's distance to the sink so that the cluster in an area where the node density is very high may have too many CMs. As a result, the cluster head may be loaded with too heavy burden and then would die faster. It is much worse in the case that the sensor nodes are deployed nonuniformly.

Research works in the literature [12–15] concentrate on how to switch off as many as possible sensor nodes for saving energy consumption while satisfying application's

requirements for coverage. In [12], the authors proposed a sleep scheduling mechanism that puts sensor nodes into sleeping mode based on coverage information. In [13], the authors proposed a sleep scheduling scheme, where the probability of a sensor node's entering sleeping state is proportional to its distance to its cluster head. However, achieving the expected coverage ratio does not ensure that it can provide persistent coverage of the monitored area. Obviously, sensor nodes do not equally contribute to the whole network coverage. Comparing with the loss of a sensor node in scarcely populated regions, the death of a sensor node in a densely populated area is not so important. In other words, if some parts of a sensor node's sensing area are not covered by any other sensor nodes, the loss of this sensor node would leave the network inefficient. Such a sensor node can be called the coverage-critical node. These algorithms select cluster heads according to residual energy or previous activity of the sensor node as cluster head, ignoring sensor node's overlapping degree. So in these algorithms, the coverage-critical nodes may become cluster heads. Thus, these cluster heads would run out of energy very fast so that the coverage lifetime would be affected. This paper proposes a new cluster head selection algorithm that takes sensor node's residual energy and overlapping degree into consideration at the same time.

Existing clustering routing algorithms [2–10, 19–21] perform the cluster construction operation periodically to balance the energy consumption between cluster head and cluster members. However, these approaches cannot improve energy balance of relaying nodes significantly. In FBR [16], it assigns the transferred data over multiple paths from sensor nodes to the sink in order to equalize the energy consumption of sensor nodes. But the cluster formation is performed only once in FBR, which will result in that the sensor nodes as cluster heads would deplete energy very fast. And in FBR, each cluster has the same cluster size; thus the energy hole will occur around the sink easier. In this paper, we propose a novel algorithm where flow-partitioned routing is applied. At the same time, it adopts unequal clustering mechanism that will help avoid the hot spots problem around the sink, thus prolonging network lifetime.

3. Network Model and Definitions

3.1. Network Model. In this paper, we adopt the same network model as in [7, 8, 20], considering $M \times L$ square units, denoted by S , in which N homogeneous sensor nodes are randomly deployed. Then the average node density ρ_0 is N/S . Each sensor node is assigned with a unique ID and has the same initial energy E^{init} . Sensor nodes cannot get location information themselves and all of them will not move after deployment. We assume that the sink is out of the sensing field and has enough energy. The sensing range of a sensor node is r . Each sensor node is able to communicate with the sink directly, having a tuned transmission radius, denoted by $d = kR$ ($k = 1, 2, 3, \dots$) where R is the minimum transmission range and we assume that R is larger than r .

The sensor nodes in the range R of sensor node i are called the neighbor nodes of sensor node i , denoted by $\text{NG}(i)$. And

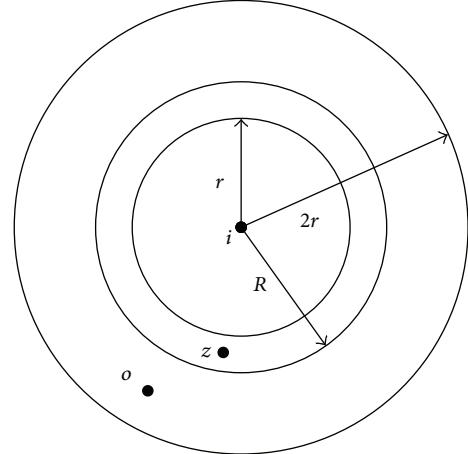


FIGURE 2: The neighbor nodes and close nodes of sensor node i .

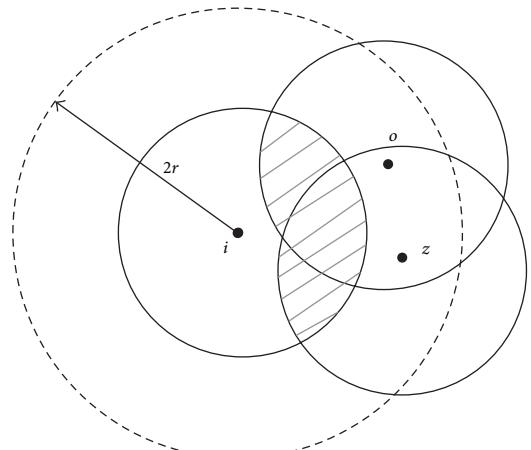


FIGURE 3: Illustration of overlapping area.

those sensor nodes in the range from r to $2r$ of sensor node i are called the close nodes of sensor node i , denoted by $\text{CN}(i)$. Figure 2 shows an example in which sensor node i has one neighbor node z and two close nodes, z and o .

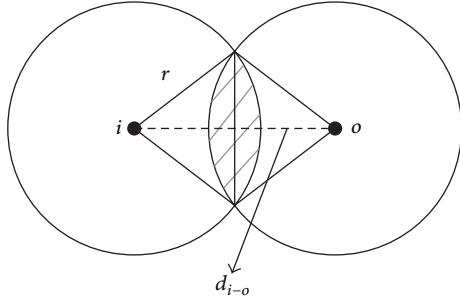
Then, $\text{NG}(i)$ and $\text{CN}(i)$ can be given as follows:

$$\begin{aligned} \text{NG}(i) &= \{j \mid d_{i-j} \leq R\}, \\ \text{CN}(i) &= \{j \mid d_{i-j} \geq r \wedge d_{i-j} \leq 2r\}. \end{aligned} \quad (1)$$

The overlapping degree of a sensor node is the ratio of the overlapping area of the sensor node with its close nodes to its sensing area. For example, the overlapping degree of sensor node i , denoted by $\text{OPD}(i)$, can be computed as follows:

$$\text{OPD}(i) = \frac{\bigcup_{x \in \text{CN}(i)} S_i \cap S_x}{S_i}. \quad (2)$$

As Figure 3 illustrates, assuming the sensing area of sensor node i is S_i and the area of shadowing portion is S' , then the overlapping degree of sensor node i is S'/S_i . Obviously, the value of $\text{OPD}(i)$ ranges from 0.0 to 1.0 and the

FIGURE 4: Sensor node i with one close node.

smaller the OPD(i) is, the more important the sensor node i to the network is. If $\text{OPD}(i) = 0$, it indicates that there are no other sensor nodes in the sensing area of sensor node i . That is to say, the information of the area covered by sensor node i cannot be got if sensor node i dies. To get $\text{OPD}(i)$, $\cup_{x \in \text{CN}(i)} S_i \cap S_x$ should be calculated first.

First of all, we consider a simple case. As Figure 4 shows, sensor node i has one close node and the sensing area of i , denoted by S_i , is πr^2 . The area of the shadow region part $\cup_{x \in \text{CN}(i)} S_i \cap S_x$ can be expressed as $S_i \cap S_o$ which can be computed according to

$$S_i \cap S_o = 2r^2 \arccos\left(\frac{d_{i-o}}{2r}\right) - \frac{d_{i-o}}{2} \sqrt{4r^2 - d_{i-o}^2}, \quad (3)$$

where r is the sensing range and d_{i-o} is the distance from sensor node i to sensor node o that can be got by the log-shadowing model [22].

So, when sensor node i has multiple close nodes, $\cup_{x \in \text{CN}(i)} S_i \cap S_x$ can be calculated according to

$$\begin{aligned} \bigcup_{x \in \text{CN}_i} S_i \cap S_x &= S_i \cap S_j + \bigcup_{x' \in \{\text{CN}(i)-j\}} S_i \cap S_{x'} \\ &\quad - \left(S_i \cap S_j \cap \left(\bigcup_{x' \in \{\text{CN}(i)-j\}} S_{x'} \right) \right), \quad (4) \\ &\quad j \in \text{CN}(i). \end{aligned}$$

3.2. Energy Model. This paper only considers the energy consumed in data transmitting and receiving as the energy model used in [2, 5–10, 20, 21]. The energy consumed for data transmitting comes from transmitter electronics and amplifier, and the energy consumed for data receiving comes from receiver electronics. Then, when transmitting one l -bit packet with distance d , the radio expends:

$$E_t(l, d) = \begin{cases} lE_{\text{elec}} + lE_{\text{fs}}d^2, & d < d_0 \\ lE_{\text{elec}} + lE_{\text{mp}}d^4, & d \geq d_0 \end{cases} \quad (5)$$

As well, when receiving one l -bit packet, the radio expends:

$$E_r(l) = lE_{\text{elec}}, \quad (6)$$

where E_{elec} is the transmitting and receiving circuit loss. The free space (d^2 power loss) is adopted when the distance between transmitter and receiver is smaller than the threshold d_0 , and the multipath model (d^4 power loss) is adopted when the distance between transmitter and receiver is larger than the threshold d_0 .

4. The Proposed Algorithm

FPUC consists of two phases: clustering and flow-partitioned routing. In the clustering phase, it first selects candidate cluster heads based on sensor node's residual energy ratio. Then it calculates competition radius according to sensor node's distance to the sink and surrounding node density. After that, CHs are generated on the basis of sensor node's residual energy and overlapping degree. In the routing phase, CH aggregates data received from CMs and then distributes the data flow to gateway nodes which will relay the received data to the next hop with minimum cost.

4.1. Cluster Formation Algorithm. At the time of network setup, the sink broadcasts a message named “Initial (t_0)” to all sensor nodes, where t_0 is the time of cluster formation. After receiving the initial message, each sensor node computes the distance to the sink according to signal strength and returns an “Init_Response” message that contains its ID and distance to the sink.

At the beginning of cluster formation phase, each sensor node broadcasts a “Compete_Head_Msg” message that includes its ID, residual energy, and distance to the sink with max competition radius R_{\max}^{comp} (assuming $R_{\max}^{\text{comp}} > 2r$ and $R_{\max}^{\text{comp}} > R$). Thus, each sensor node, say i , can find its neighbor nodes $\text{NG}(i)$ and close nodes $\text{CN}(i)$ according to all received “Compete_Head_Msg” messages. After that, sensor node i calculates its residual energy ratio (RER) according to

$$\text{RER}(i) = \frac{E_i^{\text{residual}}}{E_{\text{NG}(i)}^{\text{residual}}}, \quad (7)$$

where E_i^{residual} is the residual energy of sensor node i and $E_{\text{NG}(i)}^{\text{residual}}$ is the average residual energy of sensor nodes in $\text{NG}(i)$. If $\text{RER}(i) \geq 1$, sensor node i becomes a candidate cluster head; otherwise, it enters sleeping mode and keeps sleeping until the cluster head selection ends. If a sensor node becomes a candidate cluster head, it needs to compute its competition radius. We set R_i^{comp} as a function of the distance to the sink of sensor node i as [7, 20]

$$R_i^{\text{comp}} = \left(1 - \frac{d_{\max} - d_{i-\text{sink}}}{d_{\max} - d_{\min}} c \right) R_{\max}^{\text{comp}}, \quad (8)$$

where d_{\max} and d_{\min} denote the maximum and minimum distance between sensor nodes and the sink, $d_{i-\text{sink}}$ is the distance between sensor node i and the sink, and c is a constant parameter between 0 and 1. For a candidate cluster head i , assuming that there are N_i sensor nodes in the range

of its competition radius R_i^{comp} , the surrounding node density of candidate cluster head i can be given by

$$\rho(i) = \frac{N_i}{\pi \times R_i^{\text{comp}} \times R_i^{\text{comp}}}. \quad (9)$$

If $\rho(i) - \rho_0 > 0$, the competition radius of candidate cluster head i should be recalculated. Candidate cluster head i first counts the number of sensor nodes in its increasing range $D_0, 2D_0, 3D_0 \dots \lambda D_0$ ($\lambda D_0 \leq R_i^{\text{comp}}$; λ is the maximum positive integer that ensures that λD_0 is not larger than R_i^{comp}), denoted by $N_i(1), N_i(2), \dots, N_i(\lambda)$. Then candidate cluster head i decreases competition radius from λD_0 by the step of D_0 until the $\rho(i)$ is not larger than ρ_0 and adopts this mD_0 ($1 \leq m \leq \lambda$) as the final competition radius.

Once finishing computing competition radius, all candidate cluster heads start to compete to be cluster head. In FPUC, it adopts timing broadcast mechanism. The time that a candidate cluster head declares to become a cluster head is defined as follows:

$$t_i = \begin{cases} \mu \times \left(\alpha \times \frac{\overline{E_{\text{NC}(i)}}}{E_i^{\text{residual}}} + (1 - \alpha) \right. \\ \left. \times (1 - \text{OPD}(i)) \right) \times t_0, & E_i^{\text{residual}} \geq \overline{E_{\text{NC}(i)}} \\ +\infty, & E_i^{\text{residual}} < \overline{E_{\text{NC}(i)}}, \end{cases} \quad (10)$$

where $\text{NC}(i)$ is the candidate cluster heads in the range of R_i^{comp} of sensor node i , E_i^{residual} is the residual energy of candidate cluster head i , $\overline{E_{\text{NC}(i)}}$ is the average residual energy of sensor nodes in $\text{NC}(i)$, $\text{OPD}(i)$ is the overlapping degree of candidate cluster head i , μ is a random decimal between 0.95 and 1.0 that could reduce the probability of broadcast conflict, and α is a regulator that ranges from 0 to 1. According to (10), the more the residual energy and the larger the overlapping degree the candidate cluster head has, the shorter the waiting time to become a cluster head is. Once a candidate cluster head becomes a cluster head, it will broadcast a message named “Final_Head_Msg” to inform other sensor nodes and if a candidate cluster head receives a “Final_Head_Msg” message, it will quit the cluster head competition.

When the cluster head selection phase ends, ordinary sensor nodes wake up. To inform the ordinary sensor nodes to join cluster, each cluster head broadcasts a message named “Construct_Cluster_Msg.” After that, each ordinary node joins the closest cluster according to signal strength and returns a “Join_Cluster_Msg” to notice the corresponding cluster head.

Algorithm 1 is the pseudocode of cluster head selection algorithm.

4.2. Flow-Partitioned Routing Algorithm. In this phase, each cluster member first sends data to its cluster head. The cluster head aggregates the data gathered from its cluster members into a single packet, which is assumed L -bit, and then

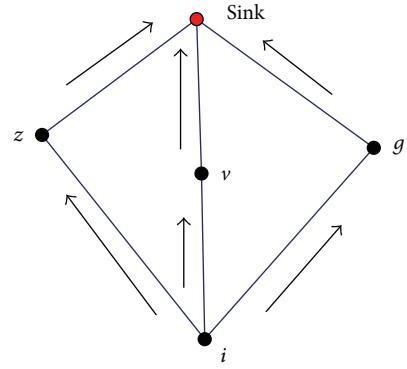


FIGURE 5: Flow-partitioned transmission.

transmits the data to the sink by the way of flow partitioning, which can significantly balance energy consumption among sensor nodes. Figure 5 illustrates an example; to explain the advantages of flow-partitioned routing more clearly and briefly, we make the following assumptions: (1) sensor node i has a Y -bit data packet, denoted by P , to be conveyed and the residual energy of sensor nodes z , v , and g is 0.4 J; (2) the energy consumption for transmitting Y -bit data from sensor nodes z , v , and g to the sink is 0.3 J and the energy consumption for receiving Y -bit data from sensor nodes z , v , and g is 0.12 J, 0.1 J, and 0.12 J. So if sensor node i sends packet P to the sink via sensor node v , sensor node i will run out of its energy. If the data packet P is partitioned into three equal packets, denoted by p_1 , p_2 , and p_3 , each of which is $Y/3$ -bit and sensor node i sends p_1 , p_2 , and p_3 to sensor nodes z , v , and g ; thus after the sink receives the three packets from sensor nodes z , v , and g , the residual energy of sensor nodes z , v , and g is 0.26 J, 0.27 J, and 0.26 J. Obviously, flow-partitioned routing can effectively balance energy consumption among sensor nodes, thus significantly prolonging the network lifetime.

The flow-partitioned routing algorithm is composed of two phases: data flow partitioning phase and relaying phase. In the data flow partitioning phase, the cluster head partitions data flow into several small packets and then distributes these packets to its gateway nodes. In the relaying phase, each gateway node transmits received data to the next hop with minimum cost. If a sensor node, say j , receives data from a gateway node, sensor node j will send the data to its cluster head.

Figure 6 shows the entire process intuitively. In the data flow partitioning phase, each cluster head, say j , first finds its gateway nodes $\text{GN}(j)$ according to

$$\begin{aligned} \text{GN}(j) = \{i \mid i \in \text{Cluster Mem}(j) \wedge E_i^{\text{residual}} \\ \geq \bar{E} \wedge d_{i-\text{sink}} \leq d_{j-\text{sink}}\}. \end{aligned} \quad (11)$$

```

(0) for every node in the network
(1) broadcast a Compete_Head_Msg
(2) for every node  $i$  in the network
(3)   if receive a Compete_Head_Msg from node  $j$  then
(4)     compute distance between node  $i$  and node  $j$ 
(5)   end if
(6) for every node  $i$  in the network
(7)   find NG( $i$ ) and CN( $i$ )
(8)   compute OPD( $i$ ) according to (2), (3), (4)
(9)   compute RER( $i$ ) according to (7)
(10)  if RER( $i$ )  $\geq 1$  then
(11)    beCandidateClusterHead  $\leftarrow$  TRUE
(12)    compute  $R^{\text{comp}}$  of node  $i$  according to (8), (9)
(13)  else
(14)    sleep
(15)  end if
(16) for every CandidateClusterHead in the network
(17)   compute  $t_i$  according to (10)
(18) for every CandidateClusterHead in the network
(19)   while (the timer  $t_0$  is not expired)
(20)     if ( $CurrentTime < t_i$ )
(21)       if receive a Final_Head_Msg
(22)         give up the competition
(23)       end if
(24)     else if ( $CurrentTime == t_i$ )
(25)       beClusterHead  $\leftarrow$  TRUE
(26)       broadcast a Final_Head_Msg
(27)     end if
(28)   end while

```

ALGORITHM 1: The pseudocode of cluster head selection algorithm.

Assuming sensor node j has H -bit data to be transmitted, the amount of data to be distributed to each of its gateway nodes, say i , is given as follows:

$$h_i = \frac{E_i^{\text{residual}}}{\sum_{v \in \text{GN}(j)} E_v^{\text{residual}}} H, \quad i \in \text{GN}(j). \quad (12)$$

When a gateway node, say i , receives data sent by its cluster head, it needs to find the next hop to forward data. In order to reduce energy consumption for communication, a distributed strategy is adopted where gateway nodes choose an optimal path for transmission. At the beginning, gateway node i should decide whether to send data to the sink directly or broadcast a message which includes its ID, its cluster head ID, and its distance to the sink depending on whether its distance to the sink $d_{i-\text{sink}}$ is larger than the threshold TD. When a sensor node, say j , receives the message from gateway node i , it will give gateway node i a response message that contains its ID, its residual energy E_j^{residual} , its distance to the sink $d_{j-\text{CH}}$, its cluster head's residual energy $E_{\text{CH}}^{\text{residual}}$, and its cluster head's distance to the sink $d_{\text{CH-sink}}$ if it is not in the same cluster as gateway node i and it is closer to the sink than gateway node i . After receiving the response message from sensor node j , gateway node i computes the distance to sensor node j , d_{i-j} and adds the information of sensor node j into the candidate relaying nodes table CRN(i). At last, gateway node

i will select the sensor node with minimum cost in CRN(i) as the next hop. The cost in this paper is defined as follows:

$$\begin{aligned} \text{cost}(i, j) = & \theta \frac{E_i^{\text{init}}}{E_j^{\text{residual}}} + \eta \frac{E_{\text{CH}}^{\text{init}}}{E_{\text{CH}}^{\text{residual}}} \\ & + \omega \frac{d_{i-j}^2 + d_{i-\text{CH}}^2 + d_{\text{CH-sink}}^2}{d_{i-\text{sink}}^2}, \end{aligned} \quad (13)$$

$$\theta + \eta + \omega = 1,$$

where θ , η , and ω are parameter factors. The cost function is mainly based on the following considerations.

- (1) The first term in the cost function aims to select the sensor node with more residual energy as relaying node to forward data. Because data transmission will consume energy, the residual energy of a sensor node is a very important factor.
- (2) The second term in the cost function takes the residual energy of candidate relaying node's cluster head into consideration because when a relaying node receives data, it will send the data to its cluster head.
- (3) The third term in the cost function aims to select the route that uses minimum energy to transmit data from sensor node i to the sink.

In the data flow partitioning phase, it may occur that some cluster heads, say j , may fail to get their gateway nodes, called “Gateway Hole.” As Figure 7 shows, each of the cluster members that are closer to the sink of cluster head j does not have more residual energy than the cluster’s average residual energy. If the problem arises, cluster head j will broadcast a message that contains its distance to the sink with δR_j^{comp} (δ is the minimum positive integer that ensures that $\text{NCH}(j)$ is not empty). When the cluster head that is closer to the sink than cluster head j , say i , receives the message, it will return a message that contains its residual energy. After receiving these messages, cluster head j will set up a neighbor cluster heads table $\text{NCH}(j)$ and add the information of cluster head i into $\text{NCH}(j)$. The amount of data distributed to each cluster head in $\text{NCH}(j)$ is given as follows:

$$h_i = \frac{E_i^{\text{residual}}}{\sum_{v \in \text{NCH}(j)} E_v^{\text{residual}}} H, \quad i \in \text{NCH}(j). \quad (14)$$

5. Performance Analysis

Lemma 1. *In the cluster formation phase, the message complexity is $O(N)$, where N is the total number of sensor nodes in the network.*

Proof. At the beginning of the cluster selection phase, each of the sensor nodes broadcasts a “Compete_Head_Msg” message, so the total number of “Compete_Head_Msg” messages is N . Assuming that in the cluster head selection phase, Q cluster heads are generated. Thus these Q cluster heads will broadcast Q “Final_Head_Msg” messages. To construct cluster, each cluster head should broadcast a “Construct_Cluster_Msg” message and each ordinary sensor node should send a “Join_Cluster_Msg” message. The number of these two types of messages is N . So the total messages add up to $N + Q + N = 2N + Q$.

Thus the message complexity is $O(N)$. \square

Lemma 2. *In cluster formation phase, the upper bound for the number of cluster heads is $\lceil 2S/\sqrt{3}((1 - c)R_{\max}^{\text{comp}})^2 \rceil$ and the lower bound is $\lceil 2S/3\sqrt{3}(R_{\max}^{\text{comp}})^2 \rceil$.*

Proof. For each cluster head, there are no other cluster heads in the range of its competition radius. In [22], it has been proved that the maximum overlapping angle between three adjacent disconnected sensor nodes is $\pi/3$. As Figure 8 shows, the number of cluster heads in the network reaches the maximum value. It can be computed that the area of a cluster is $\sqrt{3}(R^{\text{comp}})^2/2$. Figure 9 shows an opposite case. The area of a cluster reaches the maximum value $3\sqrt{3}(R^{\text{comp}})^2/2$, which has been proved in [23]. It can be easily seen that R^{comp} ranges from $(1 - c)R_{\max}^{\text{comp}}$ to R_{\max}^{comp} ; therefore, the upper bound for the number of cluster heads is $\lceil 2S/\sqrt{3}((1 - c)R_{\max}^{\text{comp}})^2 \rceil$ and the lower bound is $\lceil 2S/3\sqrt{3}(R_{\max}^{\text{comp}})^2 \rceil$. \square

Lemma 3. *The total number of messages in the cluster formation phase is smaller than $2N + \lceil 2S/\sqrt{3}((1 - c)R_{\max}^{\text{comp}})^2 \rceil$.*

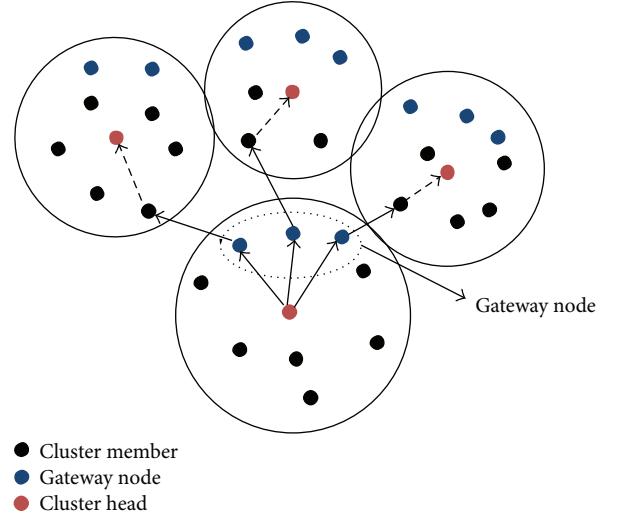


FIGURE 6: The gateway nodes of a cluster.

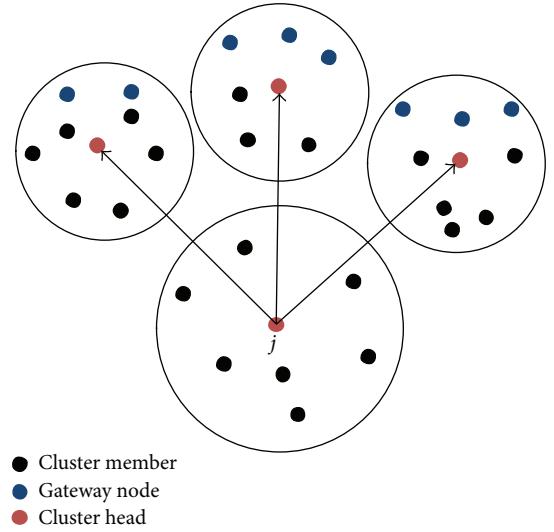


FIGURE 7: The phenomenon of gateway hole.

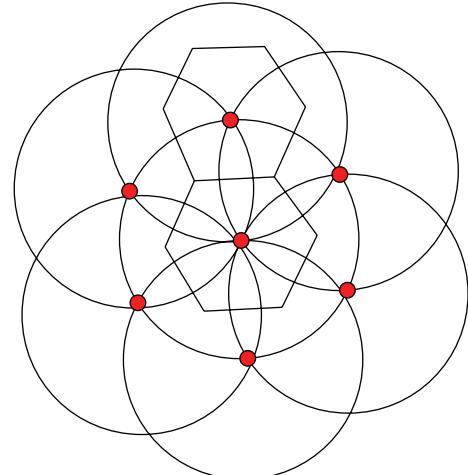


FIGURE 8: Illustration of minimum area of cluster.

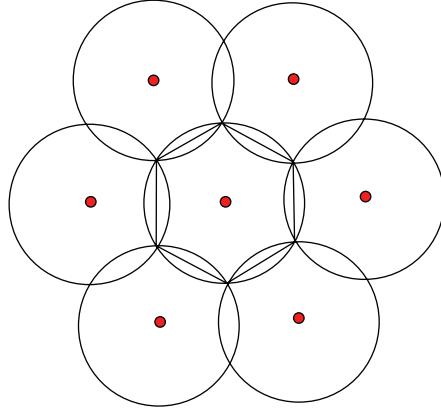


FIGURE 9: Illustration of maximum area of cluster.

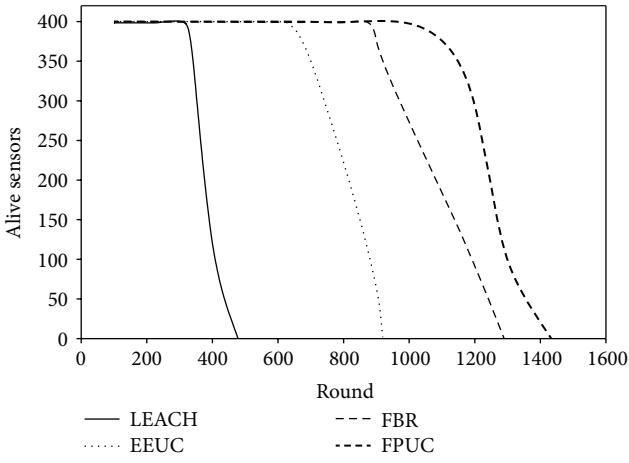


FIGURE 10: Comparison of network lifetime.

Proof. It can be concluded from Lemma 1 that the total number of messages in the cluster formation phase is $2N + Q$. Obviously, we have $Q < \lceil 2S/\sqrt{3}((1 - c)R_{\max}^{\text{comp}})^2 \rceil$; thus $2N + Q < 2N + \lceil 2S/\sqrt{3}((1 - c)R_{\max}^{\text{comp}})^2 \rceil$. \square

6. Simulation Results

In this section, we provide simulation results to validate the effectiveness of FPUC and compare the performance with existing algorithms: LEACH [5], EEUC [7], and FBR [16]. Firstly, we show two performance metrics of the four algorithms: network lifetime and coverage lifetime. Then we analyze the energy balance and overall energy consumption of each algorithm. Besides, to observe the effects of parameters on performance, we also change the network parameters to see the difference. At last, we compare the performance of the four algorithms under uniform and nonuniform deployment situations. The network environment and experimental default parameters are given in Table 1.

6.1. Comparison of Network Lifetime. In this paper, we use two metrics: network lifetime and coverage lifetime to evaluate the performance of LEACH [5], EEUC [7], FBR [16],

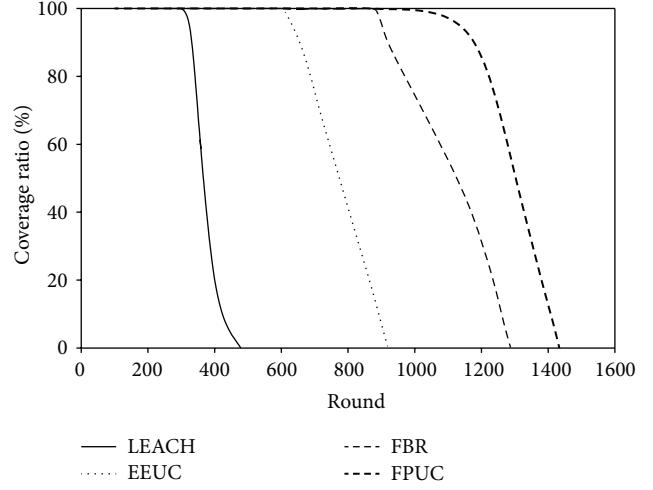


FIGURE 11: Comparison of coverage lifetime.

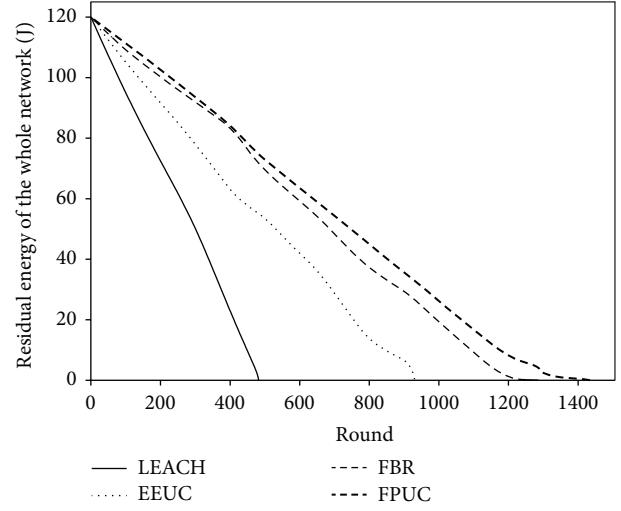


FIGURE 12: Energy consumption of the four algorithms over time.

TABLE 1: Network environment and experimental default parameters.

Parameter	Value	Parameter	Value
Network size	$200 \times 200 \text{ m}^2$	d_0	87 m
Number of nodes	400	D_0	2 m
Sink position	(100, 250)	R_{\max}^{comp}	90 m
Initial energy	0.3 J	R	30 m
Data packet size	2000 bits	r	10 m
Control msg size	100 bits	c	0.5
E_{elec}	50 nJ/bit	α	0.5
E_{fs}	10 pJ/bit/m ²	θ, η, ω	0.4, 0.4, 0.2
E_{mp}	0.0013 pJ/bit/m ⁴		

and FPUC. Network lifetime is defined as the duration from the beginning to the time when any or a given percentage of sensor nodes die. Coverage lifetime measures the period from the network setup time to the time that network

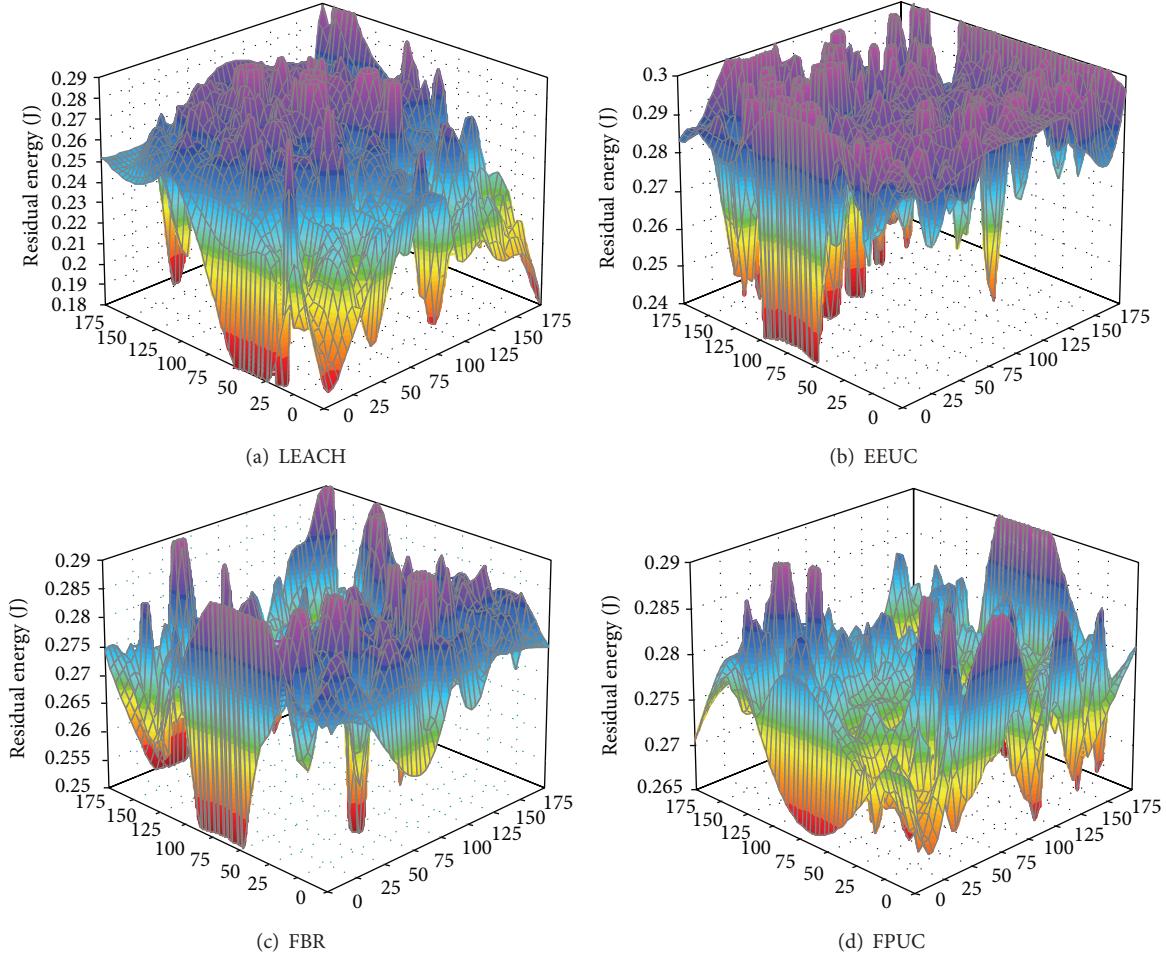


FIGURE 13: Each sensor node's residual energy of various algorithms at the 100th round.

coverage drops below a predefined threshold or application's requirement.

Figure 10 compares the network lifetime of FPUC with that of LEACH, EEUC, and FBR. From the results shown in the picture, we can see that FBR outperforms other approaches and FPUC yields a much longer network lifetime than FBR.

Figure 11 compares the coverage lifetime of FPUC along with LEACH, EEUC, and FBR. Respectively, FPUC can achieve better coverage lifetime than the other three protocols and when the first sensor node runs out of energy, the network coverage ratio is still kept at 100%.

6.2. Energy Consumption. For wireless sensor networks, the algorithms that aim to improve the lifetime need to have the following two features: (1) minimum total energy usage; (2) balanced energy consumption. Figure 12 shows the energy consumption over time of the four algorithms. Figure 13 represents each sensor node's residual energy at the 100th round of LEACH, EEUC, FBR, and FPUC.

We can have the following conclusions from Figures 12 and 13. The energy consumption in each round of LEACH and EEUC is much more than that of FBR. And FPUC

consumes relatively less energy than FBR. In Figure 13, we can clearly know each sensor node's residual energy at the 100th round of the four algorithms. Figure 13(a) is the distribution of each sensor node's residual energy of LEACH. It can be easily seen that the sensor nodes closer to the sink have much more energy than those sensor nodes far away from the sink. Figure 13(b) is the distribution of each sensor node's residual energy of EEUC. Obviously, the sensor nodes far away from the sink have much more energy than those sensor nodes closer to the sink. Figure 13(c) is the distribution of each sensor node's residual energy of FBR. Respectively, the energy consumption of FBR is relatively more balanced than LEACH and EEUC. Figure 13(d) is the distribution of each sensor node's residual energy of FPUC. Compared to LEACH, EEUC, and FBR, the total energy consumption among all sensor nodes of FPUC is more balanced. However, the sensor nodes closer to the sink still consume more energy.

The reasons for the above results can be summarized as follows. In LEACH, the data is transmitted to the sink directly; that is to say, the farther the sensor node is, the more energy the sensor node will consume when forwarding data. Thus, the sensor nodes far away from the sink will consume more energy. In EEUC, it adopts multihop data transmission,

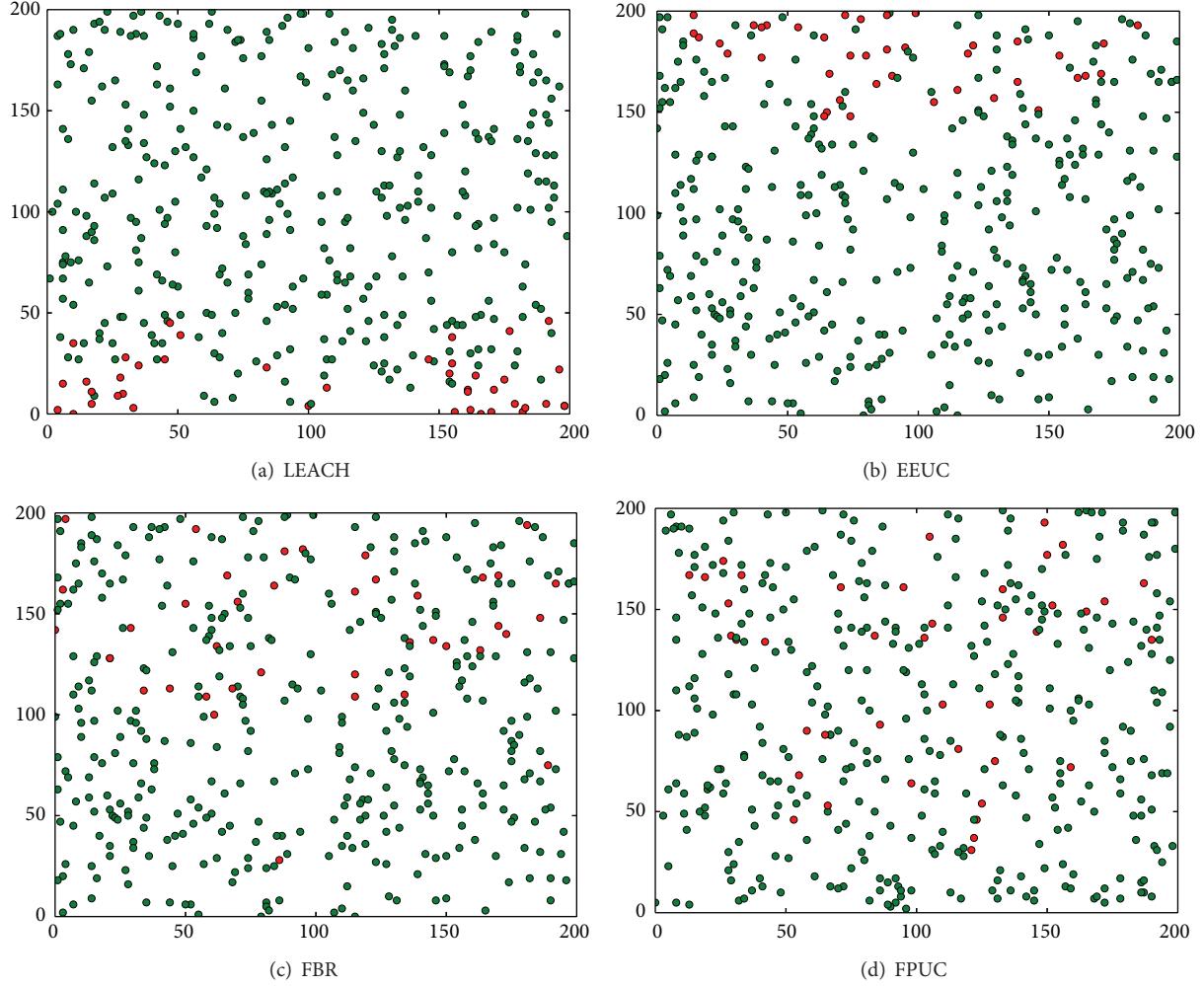


FIGURE 14: The distribution of dead sensor nodes (10% sensor nodes die).

so the sensor nodes closer to the sink will be loaded with more data forwarding traffic, thus consuming more energy. In FBR, because the flow-balanced routing is used, the energy consumption is balanced in a degree. However, because only the cluster heads convey data and the cluster formation is performed only once, the cluster heads will run out of energy faster. In FPUC, the cluster heads and cluster members share the burden of data traffic at the same time and unequal clustering is adopted, thus leading to more balanced energy consumption.

Figure 14 shows the distribution of dead sensor nodes when 10% sensor nodes die. We can see that the distribution of FPUC is more uniform in comparison with the other three algorithms.

6.3. The Effects of Network Parameters on Performance

6.3.1. Varying the Number of Sensor Nodes. First we evaluate how the number of sensor nodes affects the performance. From Figure 15, we can draw the conclusion that the number of sensor nodes has a little effect on the performance of LEACH. When the number of sensor nodes increases, EEUC,

FBR, and FPUC can achieve longer network lifetime in some degree. In EEUC, increasing the number of sensor nodes can contribute to select a more optimal relaying node when forwarding data. In FBR and FPUC, there will be more sensor nodes to share the burden of data transmission if the number of sensor nodes increases and the energy consumption will be more balanced.

6.3.2. Varying the Value of Parameter α . Figure 16 shows how the parameter α affects the network lifetime of FPUC and Figure 17 shows the effects of parameter α on the coverage lifetime. It can be seen that the performance is the best when α is 0.5.

6.3.3. Varying the Node Distribution. From the results in Figure 18, we can see that the network lifetime of LEACH, EEUC, and FBR under the uniform distribution is longer than that under the nonuniform distribution. But the distribution has a little effect on the performance of FPUC because it can also generate a reasonable topology under the nonuniform distribution.

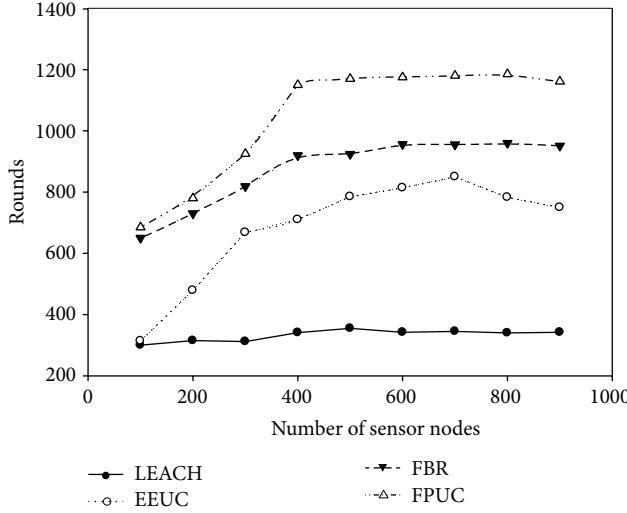


FIGURE 15: Network lifetime under different node density.

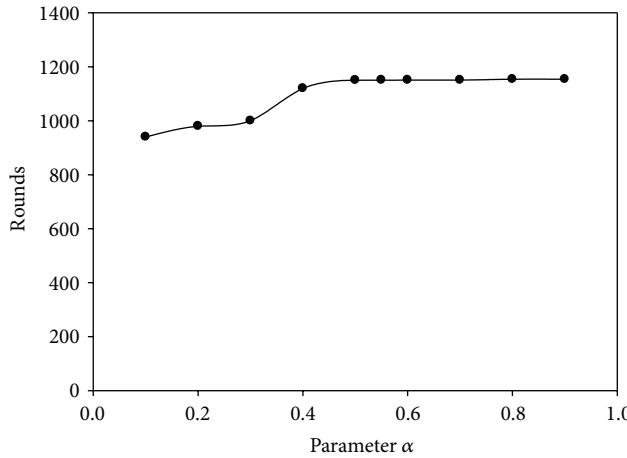
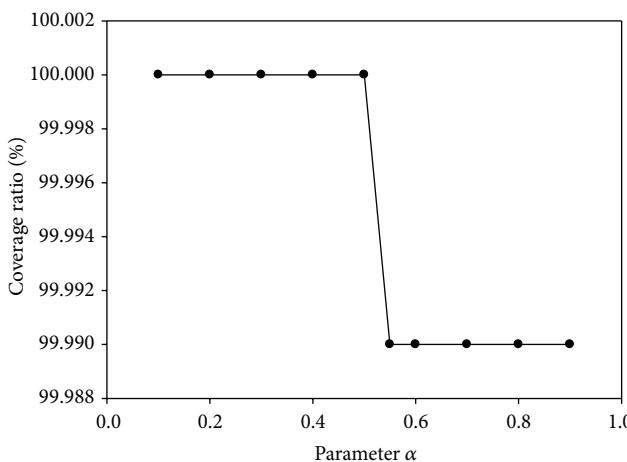
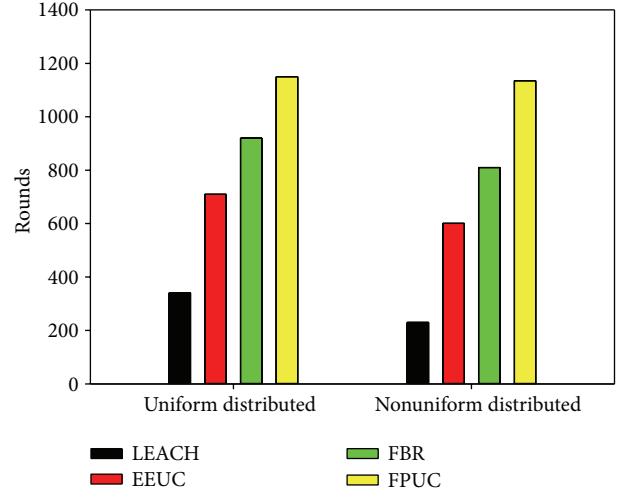
FIGURE 16: The effects of parameter α on network lifetime (when first sensor node dies).FIGURE 17: The effects of parameter α on network coverage (when first sensor node dies).

FIGURE 18: The effects of node distribution on network lifetime.

7. Summary

In this paper, we have proposed a new flow-partitioned unequal clustering routing algorithm. In the clustering formation phase, it first computes the competition radius according to the node density and the distance from a sensor node to the sink. Then it selects the sensor nodes with more residual energy and larger overlapping degree as cluster heads. In the data transmitting phase, flow-partitioned routing is adopted, where cluster heads and cluster members share the burden of data forwarding together. Comprehensive simulation results demonstrate that FPUC can significantly improve network lifetime and coverage lifetime comparing with LEACH, EEUC, and FBR.

Conflict of Interests

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61303204, U1333113), the Sichuan Province Science and Technology Support Plan (2012GZX0088-1, 2011GZ0188), and the Scientific Research Fund of Sichuan Normal University (13KYL06).

References

- [1] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.
- [2] T. Liu, Q. R. Li, and P. Liang, "An energy-balancing clustering approach for gradient-based routing in wireless sensor networks," *Computer Communications*, vol. 35, no. 17, pp. 2150–2161, 2012.
- [3] A. F. Liu, Z. H. Liu, M. Nurudeen, X. Jin, and Z. G. Chen, "An elaborate chronological and spatial analysis of energy hole for

- wireless sensor networks," *Computer Standards & Interfaces*, vol. 35, no. 1, pp. 132–149, 2013.
- [4] A. F. Liu, X. Jin, G. H. Cui, and Z. G. Chen, "Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network," *Information Sciences*, vol. 230, no. 1, pp. 197–226, 2013.
- [5] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2012.
- [6] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.
- [7] C. Li, M. Ye, G. Chen, and J. Wu, "An energy-efficient unequal clustering mechanism for wireless sensor networks," in *Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS '05)*, pp. 597–604, Washington, DC, USA, November 2005.
- [8] H. Bagci and A. Yazici, "An energy aware fuzzy approach to unequal clustering in wireless sensor networks," *Applied Soft Computing*, vol. 13, no. 4, pp. 1741–1749, 2013.
- [9] D. Kumar, T. C. Aseri, and R. B. Patel, "EEHC: energy efficient heterogeneous clustered scheme for wireless sensor networks," *Computer Communications*, vol. 32, no. 4, pp. 662–667, 2009.
- [10] D. W. Gong, Y. Y. Yang, and Z. X. Pan, "Energy-efficient clustering in lossy wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 73, no. 9, pp. 1323–1336, 2013.
- [11] W. K. Lai, C. S. Fan, and L. Y. Lin, "Arranging cluster sizes and transmission ranges for wireless sensor networks," *Information Sciences*, vol. 183, no. 1, pp. 117–131, 2012.
- [12] X. Wang, G. Xing, Y. Zhang, and C. Lu, "Integrated coverage and connectivity configuration in wireless networks," *ACM Transactions on Sensor Networks*, vol. 1, no. 1, pp. 36–72, 2005.
- [13] J. Deng, Y. S. Han, W. B. Heinzelman, and P. K. Varshney, "Scheduling sleeping nodes in high density cluster-based sensor networks," *Mobile Networks and Applications*, vol. 10, no. 6, pp. 825–835, 2005.
- [14] F. Shen, M.-T. Sun, C. Liu, and A. Salazar, "Coverage-aware sleep scheduling for cluster-based sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '09)*, pp. 1–6, April 2009.
- [15] N. Xu, A. Huang, T.-W. Hou, and H.-H. Chen, "Coverage and connectivity guaranteed topology control algorithm for cluster-based wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 1, pp. 23–32, 2012.
- [16] Y. L. Tao, Y. B. Zhang, and Y. S. Ji, "Flow-balanced routing for multi-hop clustered wireless sensor networks," *Ad Hoc Networks*, vol. 11, no. 1, pp. 541–554, 2013.
- [17] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, no. 1, article 5, 2009.
- [18] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications*, vol. 30, no. 14–15, pp. 2826–2841, 2007.
- [19] S. Soro and W. B. Heinzelman, "Prolonging the lifetime of wireless sensor networks via unequal clustering," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS '05)*, April 2005.
- [20] C. J. Jiang, W. R. Shi, X. L. Tang, P. Wang, and M. Xiang, "Energy-balanced unequal clustering routing protocol for wireless sensor networks," *Journal of Software*, vol. 23, no. 5, pp. 1222–1232, 2012.
- [21] F. J. Shang, M. Abolhasan, and T. Wysocki, "Distributed energy efficient unequal clustering algorithm for wireless sensor networks," *Journal on Communications*, vol. 30, no. 10, pp. 34–43, 2009.
- [22] A. Ahmed, "An enhanced real-time routing protocol with load distribution for mobile wireless sensor networks," *Computer Networks*, vol. 57, no. 6, pp. 1459–1473, 2013.
- [23] R. R. Choudhury and R. Kravets, *Location-Independent Coverage in Wireless Sensor Networks*, University of Illinois at Urbana-Champaign, Urbana, Ill, USA, 2004.

Research Article

An Improved Group-Based Neighbor Discovery Algorithm for Mobile Sensor Networks

Qiang Niu, Weiwei Bao, and Shixiong Xia

School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China

Correspondence should be addressed to Qiang Niu; niuq@cumt.edu.cn

Received 6 March 2014; Accepted 26 March 2014; Published 15 April 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Qiang Niu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks have been widely used in a variety of commercial and military applications, which often require neighbor nodes to establish connection quickly. Due to high mobility of nodes, it is a challenging issue to reduce discovery delay. Most discovery designs are based on pairwise and fixed duty cycle, in which discovery is passively achieved when two nodes wake up at the same time. In order to further improve the efficiency of neighbor discovery in mobile sensor networks, this paper proposes an improved group-based neighbor discovery algorithm (IGND) which dynamically adjusts the nodes' active time based on spatial properties. This paper describes the network model and the algorithm implementation in detail. Simulation results show that the improved algorithm has a good effect in reducing discovery latency and promoting discovery efficiency.

1. Introduction

With the rapid development of modern information technology, electronic equipment has evolved from the previous fixed ones to modern mobile ones and the volume becomes ever-smaller. The research directions are also moving from static sensor networks to mobile ones [1], which gradually become a research hotspot. Mobile sensor networks are mainly used in national defense, environmental monitoring, wildlife monitoring, and other aspects. In mobile sensor networks, sensor nodes first discover their neighbors constantly through communicating with each other in the moving process. Then the collected information will be forwarded.

Neighbor discovery has attracted research attention in recent years. Lots of excellent algorithms are proposed and their efficiency is also verified, such as Disco [2] and U-Connect [3]. However, previous works mainly focus on how to ensure a pair of nodes can wake up simultaneously, which has long discovery latency. Besides, the above algorithms are based on the fixed duty cycle, which brings longer discovery latency. The design goal of these traditional discovery schemes is to discover neighbors with a more energy-efficient method, no matter how long it will take, as long as it is bounded. However, in many mobile applications,

neighbor discovery has to be fast enough to enable crucial requirements, in which energy is not the most pressing concern. Thus, the need for a novel and quick discovery scheme for mobile sensor network arises.

Therefore in this paper, we advocate a group-based discovery with dynamical duty cycle. The adjusting of duty cycle is based on the potential number of neighbor nodes. An additional energy budget (in term of additional active slots) is used to perform quick neighbor discovery. By adjusting the time when nodes stay awake, more potential neighbor nodes can be discovered. Then the discovery latency can be reduced and the overall performance of the network can be improved. Specifically, our contributions are as follows.

- (i) Most of the previous work focuses on scheduling designs for pairwise discovery with fixed duty cycle. We propose the group-based discovery with dynamic duty cycle, which can reduce discovery delay with small overhead.
- (ii) By fully extracting the spatial properties, nodes' active time is adjusted based on the potential number of neighbor nodes, which can speed up the discovery process and is practical for many mobile applications.

- (iii) The performance of the proposed design is evaluated by extensive simulation studies. The results show that the discovery latency is reduced further compared with existing works.

The rest of the paper is organized as follows. Section 2 discusses the related works. Section 3 presents the network model and assumptions. Section 4 describes the basic design and makes a brief analysis of discovery delay. Section 5 presents the simulation results. Section 6 concludes the work.

2. Related Works

After years of research on the wireless sensor network, neighbor discovery is no longer a new technology, and lots of references and materials are available. The research of node discovery in which nodes always stay awake focuses on directional antenna [4, 5]; however, node discovery algorithms based on duty cycle are various, especially when the sensor nodes stay in mobile environments in which there is a high demand for discovery time, that means to find neighbor nodes as fast as possible.

There are many excellent algorithms of node discovery in mobile sensor network, such as stochastic-based protocols [6, 7], quorum-based protocols [8, 9], Disco [2], U-Connect [3], multichannel [10, 11], and conflict-based node discovery [12]. All these focus on how to ensure a pair of nodes to wake up simultaneously by a certain type of scheduling algorithm; after verification, these algorithms are effective and can ensure the nodes to discover their neighbors within a boundary of time. In stochastic-based protocols [6, 7], the nodes monitor, transfer, or sleep in stochastic cycle, getting a tradeoff compromise between the energy consumption and discovery delay. Quorum-based protocols [8] clarify the limitations on ensuring pairwise nodes to overlap in wake up time within a bounded time. Disco [2] introduces a neighbor discovery algorithm based on Chinese Remainder Theorem [13], in which the algorithm selects a pair of prime numbers as the cycle of nodes, and the selected prime numbers should meet the requirements of duty cycle. U-Connect [3] proposes a uniform neighbor discovery algorithm for symmetric and asymmetric duty cycle setting. It should be noted that U-Connect [3] for symmetric asynchronous solution is a 1.5-approximation algorithm, and the Quorum [8, 9] and Disco [2] are 2-approximation algorithms.

In traditional pairwise [2, 3, 6–12] algorithms, each node stays in active state and dormant state periodically, and the neighbor nodes in sensor networks must be in each other's communication range, only in this way we can ensure normal communication among nodes and data transmission. When two nodes schedule respectively according to their own work schedule and stay awake simultaneously, the two nodes can find each other, and add another one to their neighbor table, till now, a neighbor node is found.

Comparing with pairwise methods, group discovery [14, 15] reduces discovery delay significantly by proactively referring wake up schedules among a group of nodes. It also

builds a schedule reference mechanism among nodes to expedite the discovery process. Nevertheless, in the initialization phase of group formation, there are only a small number of known schedules. Hence, nodes cannot proactively wake up in time to verify the neighbor nodes, which may bring long discovery latency.

In all, although there exist many excellent neighbor discovery algorithms [16], most of them are pairwise methods and the discovery latency is relatively higher. There is still much room to improve based on group discovery. In this work, we introduce a novel group discovery method with dynamic duty cycle.

3. Preliminaries for Neighbor Discovery

In this section, the network model and assumptions are defined for mobile sensor networks with low duty cycle [14, 17, 18]. Some background information on how mobile nodes can discover other nodes without any infrastructure support is also introduced.

If sensor nodes always stay awake to discover neighbors in the mobile sensor networks, the nodes may be in an idle state in most of the time. Namely, there is no information for broadcasting or transmission. Although the idle state can effectively reduce energy consumption, it still wastes a large amount of energy. According to statistics, the node accounts for approximately 99% of total energy consumption while in idle preparations [19].

In order to save the limited energy of sensor nodes, sensor nodes are often set with two kinds of states: active state and dormant state. And the time is divided into continuous fixed-length time slots. The node keeps the dormant state for most of time slots and becomes active in a few slots only for sensing and communicating. Specifically, sensor nodes can broadcast and receive information packets sent by neighbors in active state, while they turn off the modules of broadcasting and receiving in the dormant state.

By controlling active and dormant status of the sensors, the energy consumption can be reduced significantly. But it will also bring additional discovery latency because the sensors can discover each other only when neighbor nodes have overlapping active slots. In order to reduce the discovery latency, the group-based [14] neighbor discovery is proposed for low duty-cycle networks.

Figure 1 shows the group discovery process from a temporal perspective. At time $t = 1$, both node B and node C are in the active state and they are in communication range of each other, then node B discovers node C at time $t = 1$, and they form a group and become aware of each other's working schedule. Similarly, at time $t = 5$, node B discovers node A and adds it to its neighbor table. At time $t = 10$, node B wakes up proactively when node A is scheduled to be active and sends the schedule of node C to node A. At time $t = 14$, node A wakes up to verify whether node C is its neighbor; if so, node A finishes the discovering of the node C. Then a new group is formed and node A continues the neighbor discovery.

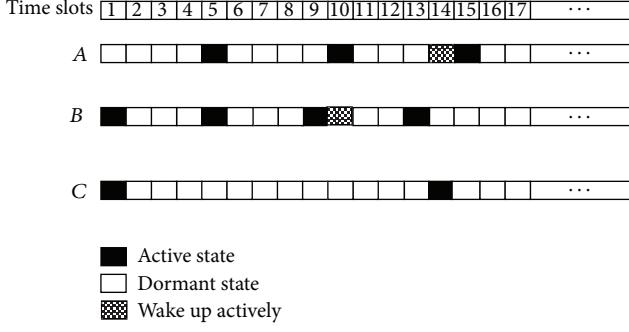


FIGURE 1: The group-based discovery process.

4. Algorithm Design

Our work is motivated by group-based discovery. The group-based discovery speeds up the discovery process by proactively referring wake up schedules among a group of nodes. The acceleration is mainly implemented from a temporal perspective. However, the embedded spatial properties are not fully considered. In fact, the position distribution of nodes in some scenarios can be obtained or estimated in advance. And the total number of nodes in a certain area can be estimated. Thus, the active time of nodes can be adjusted by comparing the number of possible total nodes and the discovered nodes by group-based algorithm. By adjusting the time when nodes stay awake, more potential neighbor nodes can be discovered. Then the discovery latency can be reduced. The following is the whole algorithm design.

Assume that the node's communication radius is R and the moving speed is v . As shown in Figure 2, the node reaches S' from S . Its moving distance is

$$\text{Distance}_{S \rightarrow S'} = v * t. \quad (1)$$

We denote the intersecting area of two communication circles as ΔS , and the area of new added region is $\Delta S'$. Consider the following:

$$\begin{aligned} \Delta S &= 2 \left(\frac{2\theta}{2} * R^2 - S_{\text{SAB}} \right) = 2(\theta * R^2 - S_{\text{SAB}}) \\ S_{\text{SAB}} &= \frac{1}{2} * 2R \sin \theta * \frac{1}{2} \text{Distance}_{S \rightarrow S'} \\ &= \frac{1}{2} R \sin \theta * \text{Distance}_{S \rightarrow S'} \\ \cos \theta &= \frac{(1/2) \text{Distance}_{S \rightarrow S'}}{R} = \frac{\text{Distance}_{S \rightarrow S'}}{2R} \\ \theta &= \arccos \frac{\text{Distance}_{S \rightarrow S'}}{2R}. \end{aligned} \quad (2)$$

From the above formula, we can obtain the area of new added region as follows:

$$\Delta S' = \pi R^2 - \Delta S. \quad (3)$$

Assume that the distribution of nodes in the network follows with λ , namely, $P \sim P(\lambda)$. The number of nodes

that will wake up is denoted as N , which is calculated based on the other known nodes' schedules in group-based discovery algorithm. According to the characteristics of Poisson distribution, we know that the expected number of nodes in new added communication area S is as follows:

$$E(S') = \lambda \Delta S'. \quad (4)$$

So the number of undiscovered nodes in the new added communication area S is

$$E(S'_{\text{no}}) = E(S') - N = \lambda \Delta S' - N. \quad (5)$$

Assume that the active time of a node, for example, node A , is T_{active} , the proactive wakeup time is $T_{\text{proactive}}$, and its period is T . We know that the ratio of the $E(S'_{\text{no}})$ undiscovered neighbors are not within the active time of the node A is

$$P_{\text{no}}^{\text{active}} = \left(1 - \frac{T_{\text{active}} + T_{\text{proactive}}}{T} \right)^{E(S'_{\text{no}})}. \quad (6)$$

In formula (6), $(T_{\text{active}} + T_{\text{proactive}})/T$ means that the ratio of one node is within the active time of the node A . Also we know that there are still some nodes that will not wake up within $T_{\text{active}} + T_{\text{proactive}}$, then the active time of the node A should be adjusted to discover the neighbors that have not been found.

Now we change the active time of the node A from T_{active} to T'_{active} .

$$T'_{\text{active}} = (T_{\text{active}} + T_{\text{proactive}})(1 + P_{\text{no}}^{\text{active}}). \quad (7)$$

According to the new active time T'_{active} , the node will additionally proactively wake up several slots to discover new potential neighbors.

As described above, the proposed algorithm is an improved group-based neighbor discovery algorithm, which fully utilizes the spatial properties. In the process of neighbor discovery, the active time of a sensor node is no longer fixed and dynamically adjusted according to the number of possible neighbor nodes. The added time slots are used to monitor and discover the neighbors in new added communication area, which can reduce the discovery latency further. The whole description of the proposed algorithm is as shown in Algorithm 1.

5. Simulation Study

In this section, we evaluate the performance of the proposed algorithm with Disco and group-based discovery algorithm through simulation. In particular, we study the impact of node density, duty cycle, radio range irregularity, and node density and moving style on system performance.

5.1. Simulation Environment. The software platform of the simulation is Windows 7 Ultimate and Visual Studio 2008. We use C++ language to simulate the algorithm and consider the scenario that 78 sensor nodes are placed on a 500 m × 500 m square field. The commercial mathematics software MATLAB is used to process the results of the simulation.

Input: The total number of sensor nodes n ; Original duty cycle DC_0 ; Time Slots table of wake-up $Table(n)$; Neighbor tables of sensor nodes $NeighborTab(n)$; Original time slot t_0 ; Current time slot t ; Next time slot t_n ; The number of discovered neighbors using group-based algorithm N .
Output: the number of discovered neighbors, discovery latency and energy consumption.

- (1) Init node parameters in mobile sensor network, including original duty cycle DC_0 , the tables of time slots $Table(n)$, neighbor tables $NeighborTab(n)$ and so on;
- (2) At time t , node i moves from last position (Old Position), begins neighbor discovery using the group-based algorithm and adds the discovered neighbors to its neighbor table $NeighborTab(i)$. Meanwhile, the number of nodes that will proactively wake up is calculated based on known other nodes' schemes, namely N ;
- (3) At time $t + 1$, node i reaches new position (New Position). It adjusts the active time according to T_{active}' , several additional slots are added to monitor and discover the potential neighbors in new area.
- (4) Continue to perform the neighbor discovery using the above method recursively.
- (5) Get the experimental data, such as discovery delay, energy consumption and so on;

ALGORITHM 1

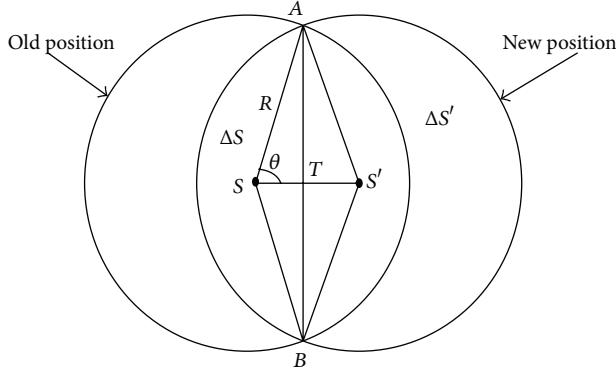


FIGURE 2: Positions of before moving and after moving.

5.2. Simulation Results

5.2.1. Impact of Node Density. The first simulation tries to investigate the impact of node density [20] on system performances. Two performance indexes are considered, which are discovery latency and energy consumption, respectively. Figure 3(a) shows the relationship between the node density and the discovery latency. From Figure 3(a), we can see that, with the density of sensor nodes increasing, the discovery latency of pairwise (Disco) algorithm increases, while the latency decreases for both the group discovery and IGND algorithms. Obviously, compared with Disco and the group discovery algorithms, our discovery algorithm IGND has the lowest discovery delay. As for energy, all three algorithms increase with the density of node increasing, and the proposed algorithm consumes more than that by the other two algorithms (Figure 3(b)).

5.2.2. Impact of Nodes Duty Cycle. The second simulation studies the impact of duty cycle on the discovery latency and energy consumption of the sensor network. As shown

in Figure 4(a), while the duty cycle increases, the discovery latency for all three algorithms decreases. It should be pointed out that Disco has significantly longer latency than the group discovery and the IGND algorithms, and the IGND algorithm has the best performance in the aspect of discovery latency. As for the energy consumption, Disco is far less than the other two algorithms, and energy consumption of the IGND algorithm is almost the same as that of the discovery design (Figure 4(b)). With the increasing of nodes' duty cycle, more active time is used to discover neighbors and communicate with each other, which will have short discovery latency but consume more energy.

5.2.3. Impact of Radio Range Irregularity. In the practical application, the radio communication range is usually irregular. This simulation studies the impact of radio range irregularity [21] on system performances. In the simulation, the radio range irregularity is assumed to change from 0 to 90%. Figures 5(a) and 5(b) show the impact of radio range irregularity on the discovery delay and the energy consumption, respectively. As shown in the above two figures, the latency and the energy consumption of all three algorithms increases when the radio range irregularity increases, and the proposed discovery design has the shortest latency with the same degree of radio irregularity. The IGND design consumes more energy than both Disco and the group discovery method, because there are additional slots to discover more neighbors.

5.2.4. Impact of Mobility Patterns. In order to reveal the impact of mobility patterns on the system performance, we simulate the system performance in the following two mobility patterns: random way and hotspot way. From Figure 6(a), we can get that the discovery latency decreases for both the two patterns with the increasing of node density. As is well known, the nodes moving in hotspot way are often clustered in some places in the network, so the nodes with this mobility

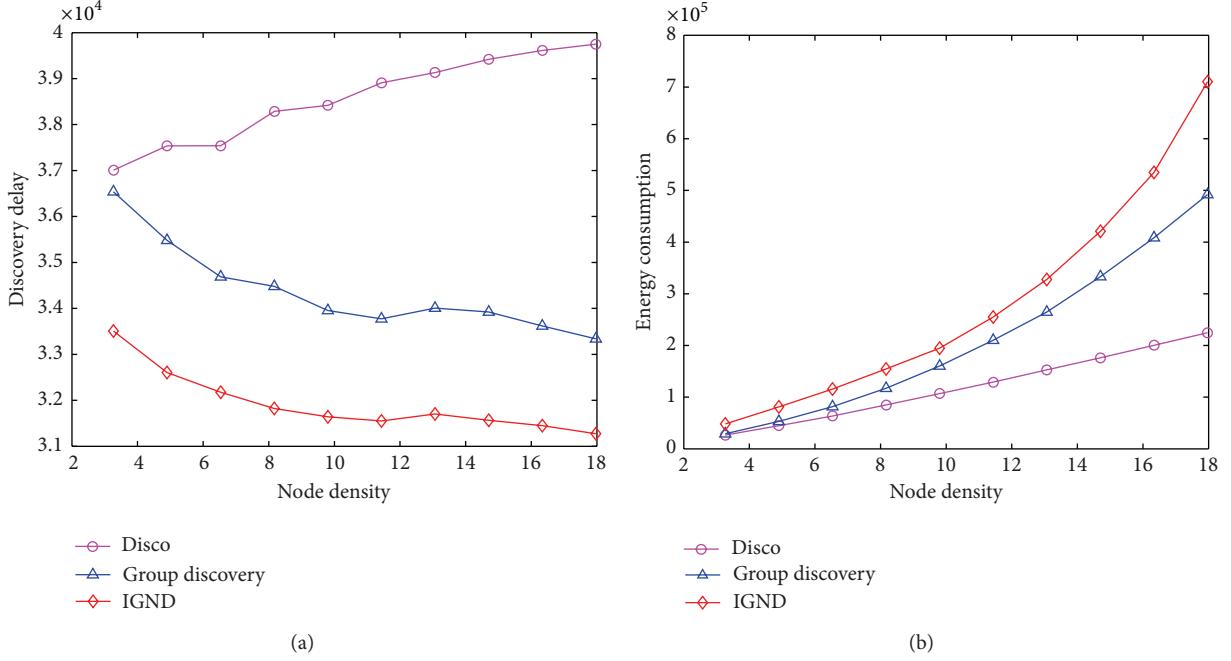


FIGURE 3: (a) Node density versus discovery delay. (b) Node density versus energy consumption.

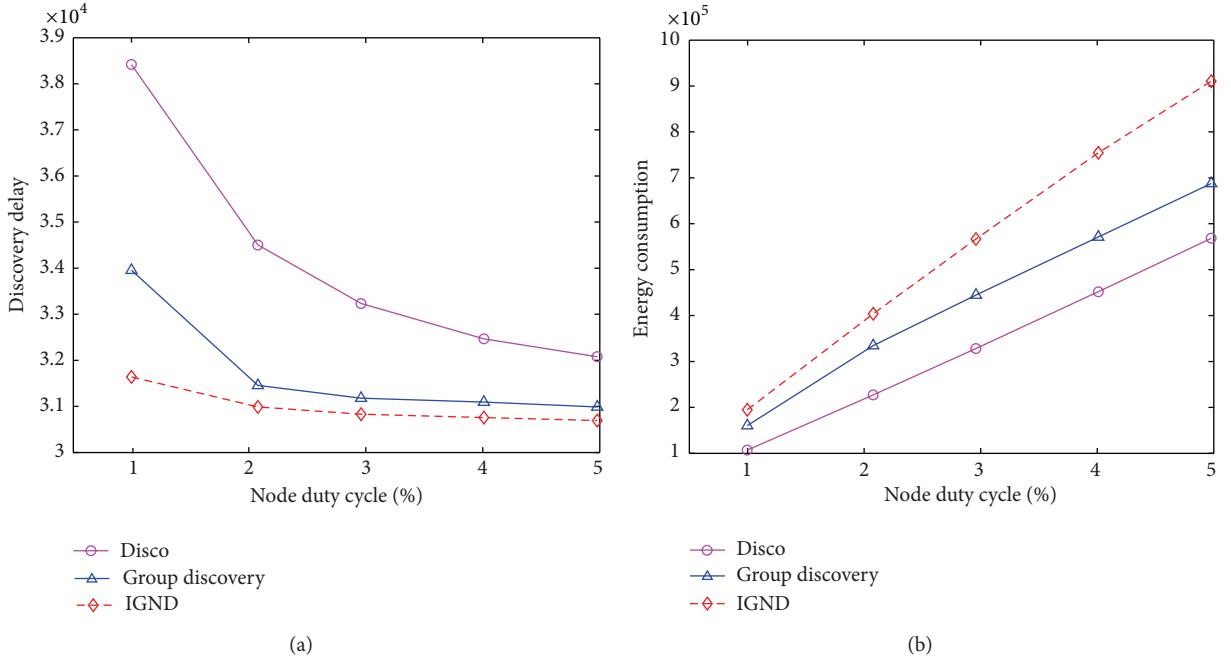


FIGURE 4: (a) Node duty cycle versus discovery delay. (b) Node duty cycle versus energy consumption.

pattern have less latency than the nodes moving in random way. However, the nodes which cluster together usually exchange data frequently; therefore, the nodes moving in hotspot way consume more energy (Figure 6(b)).

6. Conclusions

This paper proposes an improved group-based neighbor discovery, in which the active time of nodes is not fixed

and changes according to the number of potential neighbor nodes. Simulations show that the improved group-based discovery algorithm with dynamical active time has better performance in mobile sensor networks with higher node density. Although this algorithm increases the energy consumption of nodes, it significantly reduces the discovery latency of nodes. We will provide theoretical analysis of the proposed algorithm and optimize the discovery design for the underwater sensor network in the future work.

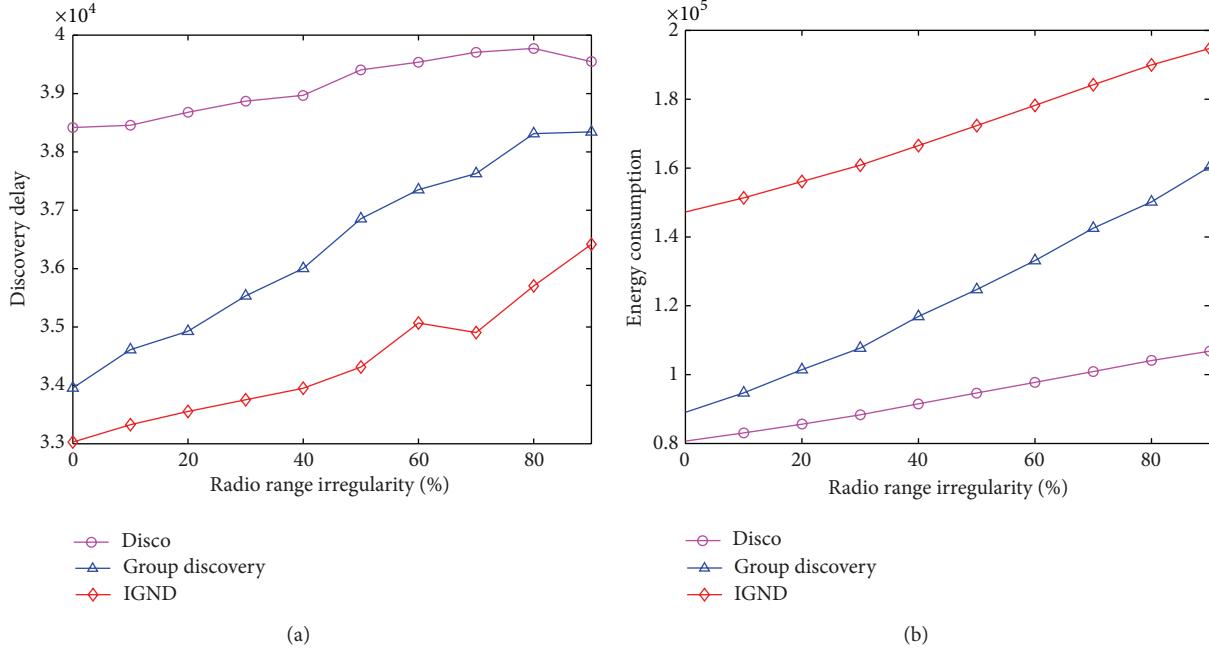


FIGURE 5: (a) Radio range irregularity versus discovery delay. (b) Radio range irregularity versus energy consumption.

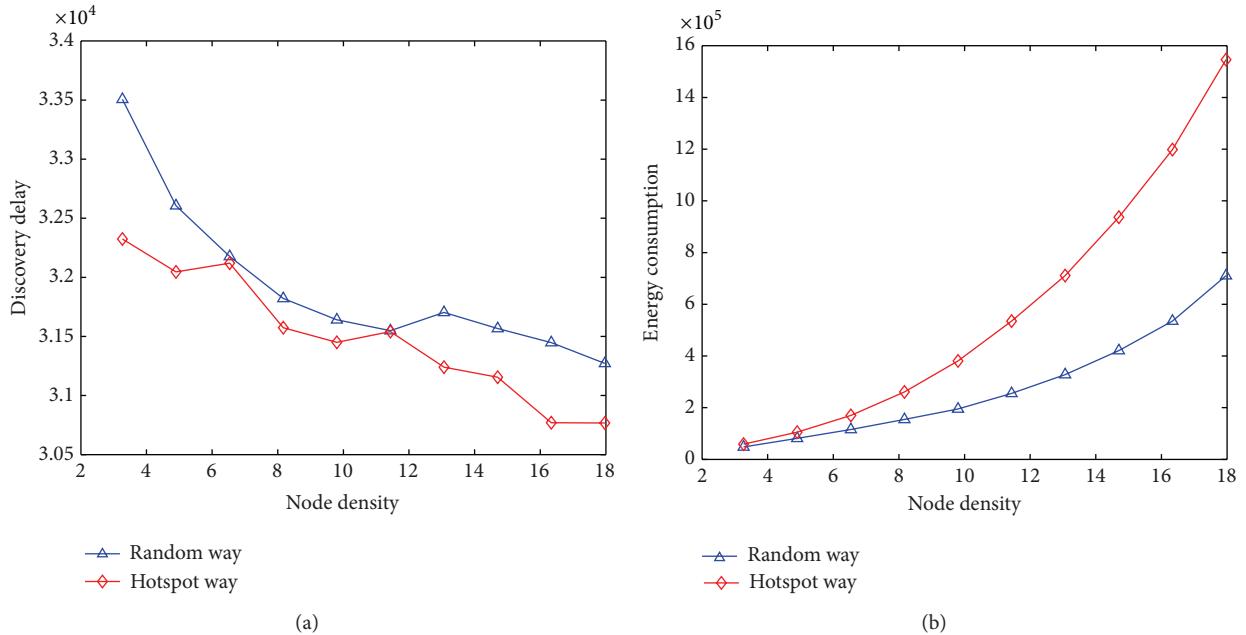


FIGURE 6: (a) Random way versus hotspot way. (b) Random way versus hotspot way.

Conflict of Interests

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

Acknowledgments

This research was supported in part by the Fundamental Research Funds for the Central Universities (2011QN23).

and the National Natural Science Foundation of China (Grant no. 61202478).

References

- [1] D. N. Alparslan and K. Sohraby, "Two-dimensional modeling and analysis of generalized random mobility models for wireless Ad hoc networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 616–629, 2007.

- [2] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications," in *Proceedings of the 6th ACM Conference on Embedded network sensor systems*, Raleigh, NC, USA, 2008.
- [3] A. Kandhalu, K. Lakshmanan, and R. Rajkumar, "U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN '10)*, pp. 350–361, Stockholm, Sweden, April 2010.
- [4] G. Jakllari, L. Wenjie, and S. V. Krishnamurthy, "An integrated neighbor discovery and MAC protocol for ad hoc networks using directional antennas," *IEEE Transactions on Wireless Communications*, vol. 6, no. 3, pp. 1114–1124, 2007.
- [5] S. Vasudevan, J. Kurose, and D. Towsley, "On neighbor discovery in wireless networks with directional antennas," in *Proceedings of the IEEE INFOCOM*, pp. 2502–2512, Miami, Fla, USA, March 2005.
- [6] S. A. Borbash, A. Ephremides, and M. J. McGlynn, "An asynchronous neighbor discovery algorithm for wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 7, pp. 998–1016, 2007.
- [7] M. J. McGlynn and S. A. Borbash, "Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '01)*, pp. 137–145, October 2001.
- [8] S. Lai, B. Ravindran, and H. Cho, "Heterogenous quorum-based wake-up scheduling in wireless sensor networks," *IEEE Transactions on Computers*, vol. 59, no. 11, pp. 1562–1575, 2010.
- [9] S. Lai, B. Zhang, B. Ravindran, and H. Cho, "Cqs-pair: cyclic quorum system pair for wakeup scheduling in wireless sensor networks," in *Principles of Distributed Systems*, vol. 5401 of *Lecture Notes in Computer Science*, pp. 295–310, 2008.
- [10] N. Karowski, A. C. Viana, and A. Wolisz, "Optimized asynchronous multi-channel neighbor discovery," in *Proceedings of the IEEE INFOCOM*, pp. 536–540, Shanghai, China, April 2011.
- [11] H.-S. Wilson So, G. Nguyen, and J. Walrand, "Practical synchronization techniques for multi-channel MAC," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 134–145, September 2006.
- [12] R. Khalili, D. L. Goeckel, D. Towsley, and A. Swami, "Neighbor discovery with reception status feedback to transmitters," in *Proceedings of the IEEE INFOCOM (InInfocom '10)*, March 2010.
- [13] I. Niven, H. S. Zuckerman, and H. L. Montgomery, *An Introduction to the Theory of Number*, John Wiley & Sons, 1991.
- [14] L. Chen, Y. Gu, S. Guo et al., "Group-based discovery in low-duty-cycle mobile sensor networks," in *Proceedings of the 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '12)*, Seoul, Republic of Korea, 2012.
- [15] D. Zhang, T. He, Y. Liu et al., "Acc: generic on-demand accelerations for neighbor discovery in mobile applications," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, Toronto, Canada, 2012.
- [16] S. Vasudevan, D. Towsley, D. Goeckel, and R. Khalili, "Neighbor discovery in wireless networks and the coupon collector's problem," in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '09)*, pp. 181–192, Beijing, China, September 2009.
- [17] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, "Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks," in *Proceedings of the IEEE Infocom*, pp. 200–209, Piscataway, NJ, USA, June 2002.
- [18] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pp. 95–107, Baltimore, Md, USA, November 2004.
- [19] T. He, P. Vicaire, T. Yant et al., "Achieving real-time target tracking using wireless sensor networks," in *Proceedings of the 12th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS '06)*, pp. 37–48, Bellevue, Wash, USA, April 2006.
- [20] X. Wu, K. N. Brown, and C. J. Sreenan, "SNIP: a sensor node-initiated probing mechanism for opportunistic data collection in sparse wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFO-COM '11)*, pp. 726–731, April 2011.
- [21] G. Zhou, T. He, S. Krishnamurthy et al., "Impact of radio irregularity on wireless sensor networks," in *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, Boston, Mass, USA, 2004.

Research Article

Mobile Beacon Based Wormhole Attackers Detection and Positioning in Wireless Sensor Networks

Honglong Chen,^{1,2,3} Wendong Chen,⁴ Zhibo Wang,⁵ Zhi Wang,⁶ and Yanjun Li⁷

¹ College of Information and Control Engineering, China University of Petroleum, Qingdao 266580, China

² The Hong Kong Polytechnic University and Shenzhen Research Institute, Shenzhen 518057, China

³ Department of Computing, The Hong Kong Polytechnic University, Kowloon 999077, Hong Kong

⁴ School of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang 330013, China

⁵ Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN 37934, USA

⁶ State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China

⁷ College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310014, China

Correspondence should be addressed to Honglong Chen; honglongchen1984@gmail.com

Received 13 February 2014; Accepted 24 February 2014; Published 30 March 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Honglong Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wormhole attack is a severe attack that can be easily launched by a pair of external attackers in hostile wireless sensor networks. In the wormhole attack, an attacker sniffs packets at one point in the network, and tunnels them through the wormhole link to the other attacker at another point of the network, which broadcasts them to its neighbors. Such kind of procedure can easily deteriorate the normal functionality of the networks. In this paper, we propose a novel wormhole attackers detection and positioning scheme based on mobile beacon, which can not only detect the existence of wormhole attacks, but also accurately localize the attackers for the system to eliminate them out of the network. The main idea is to detect whether the communication between the mobile beacon and each of the static beacons violates the communication properties and then the attacker can be estimated as the center of its communication area by determining the intersection point of the chords' perpendicular bisector. The simulation results illustrate that our proposed scheme can obtain a high wormhole attack detection probability as well as a high attackers positioning accuracy.

1. Introduction

Wireless Sensor Networks (WSNs) have been applied in more and more applications, such as the emergency response systems, military field operations, and environment monitoring systems, due to the development of the low-cost, low-power, and multifunctional sensor nodes. In normal WSN applications, the sensor nodes are organized to accomplish some kind of tasks, such as event detection or data gathering. However, since WSNs are usually deployed in hostile environments, which may be attacked by some malicious attacks, the functionality of networks may be interrupted. Therefore, security is a necessary characteristic of WSNs applications.

Attackers in WSNs can be classified into two categories, *external* attackers and *internal* attackers [1]. External attackers, such as the wormhole attack, can disrupt the network functionality without passing the system's authorization,

while internal attackers, such as the compromise attack, are authenticated ones which can act as inner-network nodes to break the system's security.

Generally, the wormhole attack is launched by two colluding external attackers, which can disrupt or even collapse the functionality of the WSNs. In the wormhole attack, an attacker sniffs packets at one point in the network and tunnels them through the wormhole link to another attacker at the other point of the network, which broadcasts the packets to its neighboring nodes. Such a simple operation can severely affect the localization and routing procedures. For example, as shown in Figure 1, two kinds of nodes, that is, beacons and sensors, are deployed in the network, which is attacked by a wormhole attack launched by A_1 and A_2 . Due to the existence of the wormhole attack, two nonneighboring nodes S_1 and S_7 will consider each other as its neighbor. Moreover, S_6 will get

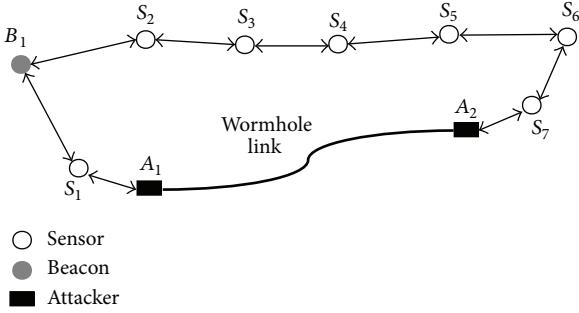


FIGURE 1: The impacts of the wormhole attack on a WSN.

a hop count 3 to B_1 via the path $S_6 \rightarrow S_7 \xrightarrow{A_2 \rightarrow A_1} S_1 \rightarrow B_1$ since the relay behaviors of A_1 and A_2 are invisible to S_1 , while the actual hop count should be 5 ($S_6 \rightarrow S_5 \rightarrow S_4 \rightarrow S_3 \rightarrow S_2 \rightarrow B_1$). Thus, the wormhole attack in Figure 1 can severely affect the self-localization of S_6 when the DV-Hop localization scheme [2] is adopted.

The above severe effects of the wormhole attack on WSN applications motivate us to propose an efficient scheme to defend against the wormhole attacks. In this paper, we first propose a novel wormhole attackers detection scheme which can detect the existence of the wormhole attacks with a high probability. Then we propose a basic positioning scheme to further localize the wormhole attackers. We also propose an enhanced positioning scheme to decrease the power consumption during the attackers positioning procedure. The main idea is to detect whether the communication between the mobile beacon and each of the static beacons violates the communication properties. The mobile beacon can localize the attacker as the center of its communication area which can be estimated by determining the intersection point of the chords' perpendicular bisector. Also, the localization accuracy of our proposed scheme is independent of the density of beacon nodes. The main contributions of this paper can be summarized as follows.

- (i) We propose a novel wormhole attacker detection scheme based on the mobile beacon and the successful detection probability is also theoretically analyzed.
- (ii) We propose a basic positioning scheme to accurately localize the wormhole attacker by estimating the center of the attacker's communication area.
- (iii) We further propose an enhanced positioning scheme for the attackers positioning to decrease the power consumption.
- (iv) We conduct simulations to illustrate the effectiveness of the proposed wormhole attackers detection and positioning schemes.

The rest of this paper is organized as follows. In Section 2, we discuss the existing wormhole attack detection and secure localization schemes. Section 3 presents the system model, including the network model and attack model. In Section 4,

we propose the wormhole attacker detection and positioning schemes. The performance evaluation is conducted in Section 5. Section 6 concludes this paper.

2. Related Work

Wormhole attack detection has been a hot research topic during the last decade and lots of schemes have been proposed. In [3], the “packet leashes” mechanism is proposed to use geographical and temporal leashes to detect whether or not the packets are attacked by wormhole attacks. The wormhole detection approach in [4] is based on the end-to-end location information. Another set of wormhole attack preventing techniques [5–7] use the round-trip time of packets as a measurement to detect whether a packet travels via the wormhole link or not. In [8], a “diameter” feature based on the local map is used to detect abnormalities caused by wormholes. LiteWorp [9] makes use of two-hop neighborhood information of the stationary network to reject the packets relayed by the wormhole attacks. MobiWorp [10] uses a secure central authority to isolate the attackers globally after they are detected.

As the wormhole attack dramatically changes the network topology, the network topology information can be used to detect the existence of the wormhole attacks. Wang et al. [11] propose to detect wormhole attacks by visualizing the entire network topology with some anomalies, which is caused by the wormhole attacks. The scheme in [12] uses the network connectivity information to detect wormhole attacks based on the observation that the number of independent neighbors of two nonneighboring nodes is upper bounded. Another connectivity-based wormhole detection approach is proposed in [13] which is robust to different communication models and energy efficient. A topological approach is proposed in [14] to detect the wormhole attacks. In [15], a localized algorithm that detects the wormhole attacks directly using the connectivity information implied by the underlying communication graph is designed, and it requires no specialized hardware, which makes it practical in the real-world scenarios. By detecting whether the communication violates the properties, some novel wormhole detection schemes are proposed in [2, 16, 17].

However, all the above wormhole attack detection schemes cannot localize the attackers, which motivate us to propose the wormhole attackers detection and positioning scheme in this paper. Our proposed scheme can achieve a higher detection probability with a satisfied attackers positioning accuracy.

3. System Model

3.1. Network Model. In this paper, we consider a WSN consisting of three types of nodes: mobile beacon, static beacons, and sensors. The mobile beacon is a node with a GPS device which moves around the network to conduct some special tasks, such as detecting the wormhole attacks. The static beacons are the nodes with fixed locations which can obtain their coordinates in advance by manual deployment

or via GPS devices. The sensors are stationary nodes in the network that initially do not know their locations. All the nodes in the network have the same transmission range, denoted as R , and we assume that there is no packet loss during the communication between any two neighboring nodes. Note that our proposed scheme can be extended to the general scenario where packet loss exists. Each of the nodes has a unique ID in the network and all of them can cooperate with each other to realize WSNs applications, such as self-localization, target tracking, and data gathering.

3.2. Attack Model. A hostile environment is considered, in which the deployed WSN will be attacked by the *wormhole attacks*. A wormhole attack is typically launched by two external attackers, which collude with each other to disrupt the network's functionality. In the wormhole attack, one attacker sniffs packets at one point in the network and forwards these packets to the other attacker via a *wormhole link*, which will immediately broadcast them to its neighboring nodes. The communication range between the attackers and the nodes is also assumed to be R . However, the communication between two colluding attackers is considered to be symmetrical and not limited to the transmission range R since the wormhole link may be based on some certain communication technique, such as the wired communication. For simplicity, we assume that the distance between each pair of wormhole attackers is larger than $2R$; that is, each pair of wormhole attackers has no communication overlapping with each other. And also, each node in the network is assumed to be covered by at most one attacker.

We denote the mobile beacon, static beacons, sensors, and attackers as MB , B , S , and A , respectively. We also denote a disk centered at u with radius R as $D_R(u)$. For example, $D_R(B_1)$ indicates the communication area of static beacon B_1 .

4. Wormhole Attacker Detection and Positioning

In this section, we will first propose the wormhole attacker detection scheme based on the mobile beacon, the detection probability of which will be theoretically analyzed. After that we will propose the wormhole attacker positioning schemes including the basic scheme and enhanced scheme, which can accurately localize the wormhole attackers.

4.1. Wormhole Attacker Detection Scheme

4.1.1. Communication Properties. Before proposing the wormhole attacker detection scheme, we firstly introduce two communication properties, which were proposed in [2] and will be the basis of the detection scheme in this paper.

Packet Uniqueness Property. A node normally cannot receive more than one copy of the same message from any of its neighbors.

Transmission Constraint Property. A node normally cannot communicate with nodes outside its transmission range.

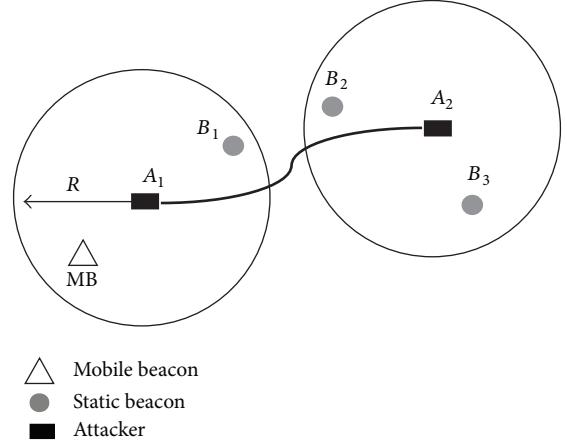


FIGURE 2: Communication procedure among the nodes under the wormhole attack.

In WSNs, the nodes may need to communicate with its neighbors to accomplish some tasks, and the communication in which should follow the above properties. For example, when a node sends a request message to one of its neighbors to get some information, such as the sensed temperature, it is considered to be normal that this neighbor sends a reply message to the sender for only once if there is no necessity of retransmission. Also, if two nodes are out of the transmission range of each other, it is considered to be normal that the packets from one of them cannot be directly received by the other.

4.1.2. Wormhole Attacker Detection Procedure. If a wormhole attack exists to disrupt the network's functionality, the packet uniqueness property and transmission constraint property may be violated by the nodes' communication. As shown in Figure 2, there are several nodes covered by the wormhole attack; that is, $MB, B_1 \in D_R(A_1)$ and $B_2, B_3 \in D_R(A_2)$. The distance between B_1 and B_2 is shorter than R and the distance between B_1 and B_3 is longer than R . When B_1 sends a request message to its neighbors, every node which receives the request message will send a reply message to B_1 . The reply message from B_2 will arrive at B_1 twice, one directly from B_2 to B_1 since their distance is shorter than R and the other via the path $B_2 \rightarrow A_2 \rightarrow A_1 \rightarrow B_1$. Thus, it violates the packet uniqueness property. Although B_3 is out of the transmission range of B_1 , it can also receive the request message via the path $B_1 \rightarrow A_1 \rightarrow A_2 \rightarrow B_3$. Similarly, the reply message from B_3 can also be received by B_1 . Since the distance between B_1 and B_3 is longer than R , their communication violates the transmission constraint property.

To detect the wormhole attackers in the network, we can use a mobile beacon to detect whether the communication between itself and each of its neighboring static beacons violates the above communication properties. The wormhole attackers detection procedure is as follows.

- (i) The mobile beacon moves in the network with some direction and step length and then stops to get its current location using GPS. The details of the

- mobile beacon's mobility model including the moving direction and step length will be discussed in the wormhole attackers positioning scheme.
- (ii) At each stopping location, the mobile beacon broadcasts a request message Req to its neighboring static beacons. Each static beacon who receives the request message will immediately reply a Rep message, including its ID and coordinate, to the mobile beacon.
 - (iii) When receiving the Rep message from the static beacon, the mobile beacon will check the receiving times of the Rep from each of the static beacons. If it receives a Rep message from a static beacon more than once, it can determine that there is a wormhole attacker within its transmission range.
 - (iv) If the mobile beacon receives the Rep message from each of the static beacons only once, it can then calculate the Euclidean distance between itself and each of them since the received Rep message includes the replier's coordinates. If the transmission constraint property is violated, it can determine that there is a wormhole attacker within its transmission range.
 - (v) If nothing abnormal is detected, the mobile beacon will move to next location and perform the above wormhole attacker detection scheme until it finishes the detection of the whole network.

4.1.3. Analysis of Wormhole Attacker Detection Probability. By carefully designing the mobility model of the mobile beacon, we can guarantee that the mobile beacon will move across the communication area of each attacker. Then we can get the following theorem.

Theorem 1. *The mobile beacon can detect the existence of the wormhole attack if at least one static beacon lies in the transmission range of either of the two attackers.*

Proof. Without loss of generality, we assume that there is only a static beacon B_1 in the transmission range of attacker A_1 , that is, $B_1 \in D_R(A_1)$, and no static beacon exists in the transmission range of A_2 . When the mobile beacon moves inside $D_R(A_2)$ and broadcasts a Req message, it can receive a Rep message from B_1 via the wormhole link. If the distance between the mobile beacon and B_1 is longer than R , it can determine that their communication procedure violates the transmission constraint property. Otherwise, if their distance is shorter than R , the mobile beacon will receive the Rep message from B_1 twice, one directly from B_1 to itself and the other via the wormhole link. Thus, it can determine that their communication procedure violates the packet uniqueness property. Therefore, the mobile beacon can detect the wormhole attack. Similarly, if there is only a static beacon B_1 in the transmission range of A_2 and no static beacon exists in the transmission range of A_1 , the mobile beacon can also detect the wormhole attack when it moves inside $D_R(A_1)$. \square

Based on Theorem 1, we can easily analyze the wormhole attacker detection probability of our proposed scheme. We

first consider the scenario with single wormhole attack. The probability that at least one static beacon lies in $D_R(A_1)$ (or $D_R(A_2)$) is denoted as $\Pr(A_1)$ (or $\Pr(A_2)$). Then, the probability that the mobile beacon can successfully detect the wormhole attack, denoted as Ps , can be calculated as

$$\text{Ps} = 1 - \overline{\Pr(A_1)} \cdot \overline{\Pr(A_2)}. \quad (1)$$

Assume that the deployment of static beacons follows the Poisson distribution with the density ρ_B . That is, the probability of k static beacons in an area D can be obtained as $\Pr(N_B = k) = ((D\rho_B)^k / k!) e^{-D\rho_B}$. Thus, we can get $\Pr(A_1) = \Pr(A_2) = 1 - e^{-\pi R^2 \rho_B}$. So the probability that the mobile beacon can successfully detect the wormhole attack is

$$\text{Ps} = 1 - e^{-\pi R^2 \rho_B} \cdot e^{-\pi R^2 \rho_B} = 1 - e^{-2\pi R^2 \rho_B}. \quad (2)$$

For the scenario with multiple wormhole attacks, we consider the detection to be successful only if the mobile beacon can detect all of the wormhole attacks. Since we consider the case that a static beacon can be attacked by at most one attacker, the detection of one of the multiple wormhole attacks is independent of that of other wormhole attacks. Thus, the detection probability that all the wormhole attacks can be successfully detected can be easily obtained as

$$\text{Ps} = \left(1 - e^{-2\pi R^2 \rho_B}\right)^n, \quad (3)$$

where n represents the number of wormhole attacks in the network. Note that a wormhole attack is composed of a pair of attackers.

4.2. Wormhole Attackers Positioning Schemes. Based on the above wormhole detection scheme, the mobile beacon can easily detect the wormhole attacks in the network. However, it is not enough to secure a WSN application. Thus, the mobile beacon has to further localize the attackers and then report to the system to eliminate them. In this section, we will first propose a basic positioning scheme which can localize the wormhole attackers accurately. To further reduce the power consumption during the positioning procedure, we will then propose an enhanced positioning scheme.

4.2.1. Basic Positioning Scheme. For ease of description, we assume that the WSN is deployed in a rectangular area of $d_x \times d_y \text{ m}^2$ and the origin of the coordinate plane locates at the left bottom of the area. As shown in Figure 3(a), the mobile beacon can horizontally move from the origin of the coordinate plane towards the right boundary of the area with a constant moving step length ΔL_H . It will conduct the wormhole attacker detection scheme each time it moves for a distance of ΔL_H . When the mobile beacon arrives at the right boundary, it can move up for a distance of ΔL_P and then move horizontally towards the left boundary with a step length of ΔL_H . Similarly, when the mobile beacon arrives at the left boundary, it will move up for a distance of ΔL_P and then move horizontally towards the right boundary of the area with a step length of ΔL_H . Such operations will be conducted until the whole area is completely scanned by the mobile beacon.

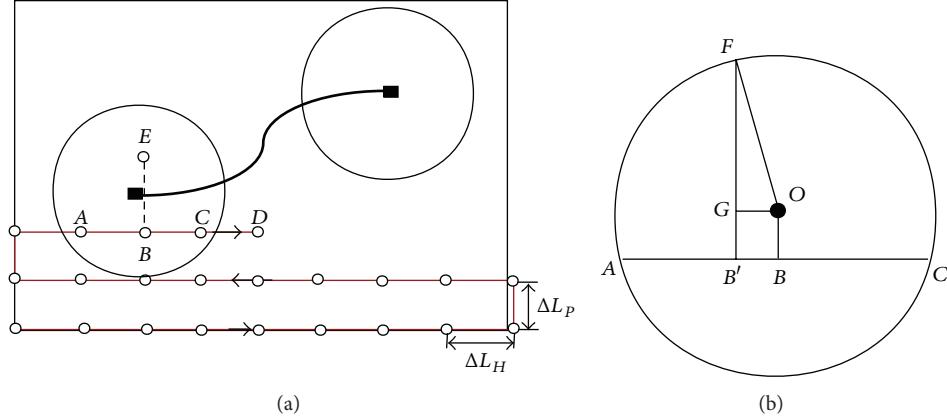


FIGURE 3: Attackers positioning of the basic scheme: (a) the attackers positioning procedure; (b) determination of ΔL_P .

When the mobile beacon enters the communication area of an attacker, such as A in Figure 3(a), it can detect the existence of the wormhole attack. Then it can save its current location as $v_f = (x_A, y_A)$. After that it continues to move towards the right boundary and can detect the wormhole attack at B and C . When the mobile beacon arrives at D , it cannot detect the wormhole attack anymore, and then it can save its previous location (C in Figure 3(a)) as $v_i = (x_C, y_C)$. After that the mobile beacon can estimate the line AC as the chord of the attacker's communication area and the attacker should lie on the perpendicular bisector of AC ; that is, the location of the attacker should be $((x_A + x_C)/2, (y_A + y_C)/2 - \sqrt{R^2 - (x_A - x_C)^2/4})$ or $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$. To further determine the actual location of the attacker, it can then move to $((x_A + x_C)/2, (y_A + y_C)/2 + R)$ (E in Figure 3(a)) and check whether it can detect the wormhole attack. If yes, it indicates that the attacker lies above line AC , and the mobile beacon will estimate the attacker's location as $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$; otherwise, the attacker's location is $((x_A + x_C)/2, (y_A + y_C)/2 - \sqrt{R^2 - (x_A - x_C)^2/4})$.

Note that the mobile beacon only conducts the wormhole attack detection scheme every step of ΔL_H , and it cannot exactly determine the intersection points between the chord and the attacker's communication circle. As shown in Figure 3(a), A and C are not exactly on the attacker's communication circle. Thus, we need to carefully design ΔL_H and ΔL_P to make the proposed basic attacker positioning scheme feasible. As the value of ΔL_H determines the localization accuracy of the basic scheme, we can firstly set it as $\Delta L_H = \alpha R$, where $0 < \alpha < 1$. Then we have to determine the value of ΔL_P . As shown in Figure 3(b), when mobile beacon moves from A to C , it will estimate the location of the chord AC 's midpoint with a maximum error of $\Delta L_H/2$. Assume that the mobile beacon estimates it as B' , and then $BB' \leq \Delta L_H/2$. To guarantee that the mobile beacon can still detect the wormhole attack when it moves to $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$, it must be satisfied that $B'F \geq R$. As $BB' \leq \Delta L_H/2$, we can get

$$B'F = GF + B'G \geq \sqrt{R^2 - \frac{\Delta L_H^2}{4}} + B'G. \quad (4)$$

Then to guarantee that $B'F \geq R$, we can make $\sqrt{R^2 - \Delta L_H^2/4} + B'G \geq R$. Finally, we can get $B'G \geq R - \sqrt{R^2 - \Delta L_H^2/4}$. Also, to guarantee that the mobile beacon can conduct the wormhole attack for at least once on line AC , it must satisfy that $AC \geq \Delta L_H$. So $B'G = OB \leq \sqrt{R^2 - \Delta L_H^2/4}$. Thus, to guarantee that the mobile beacon can correctly localize the attacker when it moves along AC , it must satisfy that

$$R - \sqrt{R^2 - \frac{\Delta L_H^2}{4}} \leq B'G \leq \sqrt{R^2 - \frac{\Delta L_H^2}{4}}. \quad (5)$$

To guarantee that the mobile beacon can move across the attacker's communication range with the condition that $B'G \in [R - \sqrt{R^2 - \Delta L_H^2/4}, \sqrt{R^2 - \Delta L_H^2/4}]$, it must satisfy that $\Delta L_P \leq \sqrt{R^2 - \Delta L_H^2/4} - (R - \sqrt{R^2 - \Delta L_H^2/4})$. That is,

$$\Delta L_P \leq 2\sqrt{R^2 - \frac{\Delta L_H^2}{4}} - R. \quad (6)$$

The previous scheme can localize the attackers accurately when there exists only one wormhole attack. However, if multiple wormhole attacks exist and the attackers are close enough to each other, there may be some problem. For example, as shown in Figure 4, when the mobile beacon node moves from C to D , although it is out of transmission range of attacker A_3 , it can still detect the wormhole attack since the communication between itself and B_1 violates the communication property. Thus, using the previous scheme, the mobile beacon node will incorrectly estimate the midpoint of the chord of $D_R(A_3)$, leading to inaccurate attackers positioning.

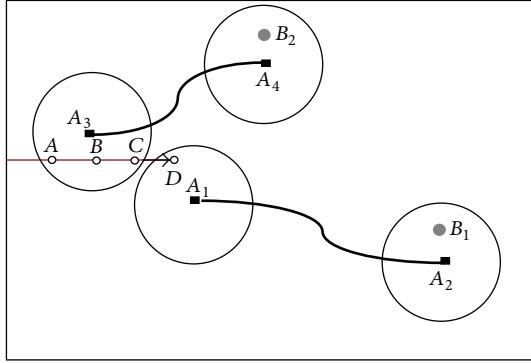


FIGURE 4: Attackers positioning of multiple wormhole attacks using conflicting set.

To solve this problem, we can use the concept of *conflicting nodes*, which we borrow from our previous work [18].

In this paper, we define the conflicting nodes of the mobile beacon node as the beacon nodes, and the communication between which and the mobile beacon node violates the packet uniqueness property or the transmission constraint property. That is to say, the conflicting nodes of the mobile beacon node are the beacon nodes which communicate with itself via the wormhole link. For example, as shown in Figure 4, when the mobile beacon node arrives at A , it can detect the wormhole attack as it can check that the communication between itself and B_2 violates the communication property; thus it will consider B_2 as its conflicting node. While it moves to D , it cannot communicate with B_2 , but it can still check that the communication between itself and B_1 violates the communication property, so it will then consider B_1 as its conflicting node.

Thus, when the mobile beacon node enters the communication range of an attacker, we can get the following theorem.

Theorem 2. *When the mobile beacon node enters the area of $D_R(A_1)$, its conflicting nodes are all the beacon nodes inside $D_R(A_2)$, where A_1 and A_2 are a pair of wormhole attackers.*

Proof. All the beacon nodes inside $D_R(A_2)$ can be classified into two sets according to the distance between each of them and the mobile beacon node. For the beacon nodes, the distance between each of which and the mobile beacon node is not larger than R , and the communication between them and the mobile beacon node violates the packet uniqueness property. Thus, the mobile beacon node will consider them as its conflicting nodes. While, for the beacon nodes, the distance between each of which and the mobile beacon node is larger than R , the communication between them and the mobile beacon node violates the transmission constraint property since they can communicate with each other via the wormhole link. Thus, the mobile beacon node will also consider these nodes as its conflicting nodes. While for other beacon nodes outside $D_R(A_2)$, since they cannot communicate with the mobile beacon node via the wormhole

link, they will not be considered as the mobile beacon node's conflicting nodes. \square

Based on Theorem 2, we can get the following corollary.

Corollary 3. *If the mobile beacon node has the same conflicting nodes at two locations, then it is within the communication range of the same attacker.*

According to Corollary 3, the mobile beacon node can identify different wormhole attackers easily. For example, as shown in Figure 4, when the mobile beacon node moves to A , it will detect the wormhole attack. Furthermore, it will consider B_2 as its conflicting node. Then when it moves to B , it can still detect the wormhole attack and B_2 is also considered as its conflicting node. Thus the mobile beacon node continues to move to C and conducts the corresponding detection. When it moves to D , although it can detect the wormhole attack, it will find that its current conflicting node is different from the previous point; thus it can determine that it moves out the transmission range of the last attacker, that is, A_3 . After that it can estimate the line AC as the chord of $D_R(A_3)$ and A_3 should lie on the perpendicular bisector of AC ; that is, the location of A_3 should be $((x_A + x_C)/2, (y_A + y_C)/2 - \sqrt{R^2 - (x_A - x_C)^2/4}$ or $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4}$. Then the mobile beacon node can first move to $((x_A + x_C)/2, (y_A + y_C)/2 - \sqrt{R^2 - (x_A - x_C)^2/4})$ to check whether it can detect the wormhole attack and also whether the conflicting node is still B_2 ; if yes, it can determine that the location of A_3 is $((x_A + x_C)/2, (y_A + y_C)/2 - \sqrt{R^2 - (x_A - x_C)^2/4}$. Otherwise, the location of A_3 is $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$.

4.2.2. Enhanced Positioning Scheme. In the basic positioning scheme, when the mobile beacon node moves forward a step, it will conduct the wormhole attack detection, in which it broadcasts a Req message to its neighboring beacon nodes and then waits for the Rep messages from them. Such kind of procedure involves the message exchange between the mobile beacon node and its neighboring beacon nodes, which is energy consuming. In this section, we will propose an enhanced positioning scheme, which can reduce the energy consumption of the positioning procedure.

Since the mobile beacon node moves across the network, it may enter the transmission range of an attacker for more than once. As shown in Figure 5, when the mobile beacon node moves from A to C , it can detect the existence of the wormhole attack after which it can determine the location of the attacker as $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$. Then it can save the currently estimated location of the attacker into the set X_A . When the mobile beacon node moves out of the transmission range of the current attacker, it will first check whether the next location lies in the transmission range of any of the attackers in X_A . If yes, it can ignore this location and check next location until the one which is outside the transmission range

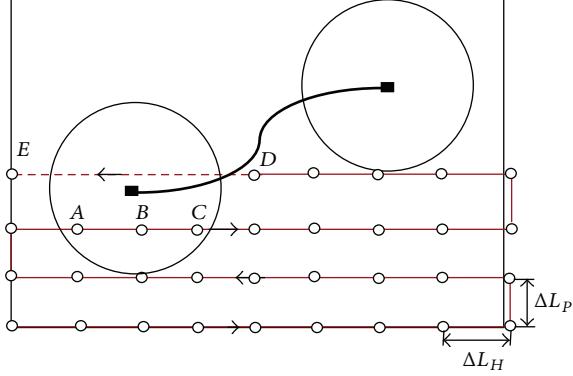


FIGURE 5: The procedure of the enhanced attackers positioning scheme.

of each of the attackers in X_A , and then it can move to that location and conduct the wormhole attack detection. Note that the attackers in X_A have already been positioned by the mobile beacon node; thus it will not affect the attacker positioning by ignoring the detection procedures at such locations. For example, as shown in Figure 5, when the mobile beacon node moves to D , it detects that there is no wormhole attack, and then it decides to move to next location, which is $(x_D + \Delta L_H, y_D)$. However, it can detect that the distance between next location and $((x_A + x_C)/2, (y_A + y_C)/2 + \sqrt{R^2 - (x_A - x_C)^2/4})$, which is estimated as the location of an attacker and is saved in X_A , is shorter than R , and then it will not conduct wormhole attack detection at this location. Similarly, the mobile beacon node will check the next several locations and find that E is not within the transmission range of each of the attackers in X_A . Thus it will directly move to E and conduct the wormhole attack detection there. By using such strategy, some unnecessary locations can be found, at which the energy-consuming wormhole attack detection procedure will not be conducted. Moreover, the enhanced positioning scheme will not degrade the attackers positioning performance.

5. Performance Evaluation

In this section, we present the simulation results to demonstrate the effectiveness of the proposed wormhole attackers detection and positioning scheme. We compare the proposed scheme with the label-based scheme proposed in [2] as the detection scheme of the label-based scheme is similar to our proposed scheme and it can also estimate the locations of the wormhole attackers. In the simulation, a WSN is randomly deployed in a $1 \times 1 \text{ km}^2$ region and the transmission range of each node is set as 150 m.

Figure 6 illustrates the comparison of the wormhole attack detection probability of our proposed scheme between the simulation and the theoretical model under single wormhole attack. We vary the number of beacon nodes from 10 to 30; that is, the density of the beacon nodes varies from 10^{-5} to 3×10^{-5} . The results show that the theoretical model matches the simulation quite well, which validates the correctness of

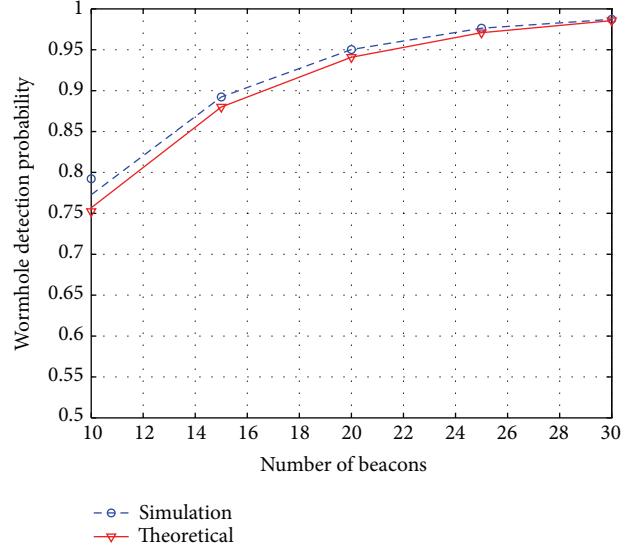


FIGURE 6: Wormhole attack detection probability: simulation versus theoretical model.

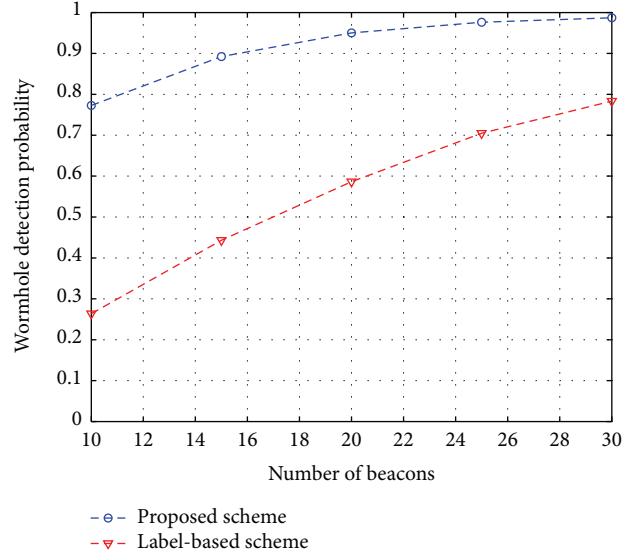


FIGURE 7: Wormhole detection probability of the proposed scheme and the label-based scheme with only single wormhole attack.

the theoretical analysis on the wormhole attack detection probability.

Figure 7 illustrates the wormhole detection probability of the proposed scheme and the label-based scheme when there is only a wormhole attack in the network. In the label-based scheme, the wormhole attack can be detected by checking whether the communications between the beacon nodes violate the properties and the precondition is that there exists at least one beacon node within the communication range of each of the attackers. In our proposed scheme, we select $\Delta L_H = 0.1R = 15 \text{ m}$, and ΔL_P is set as the maximum value corresponding to the selected ΔL_H ; that is,

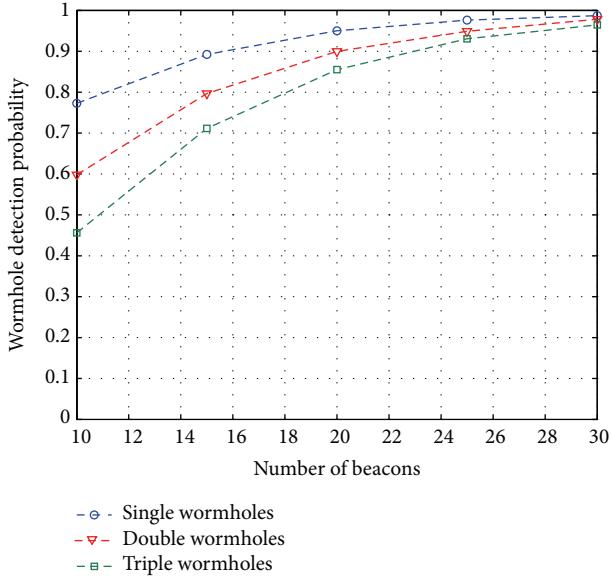


FIGURE 8: The effects of the number of wormholes on the detection probability.

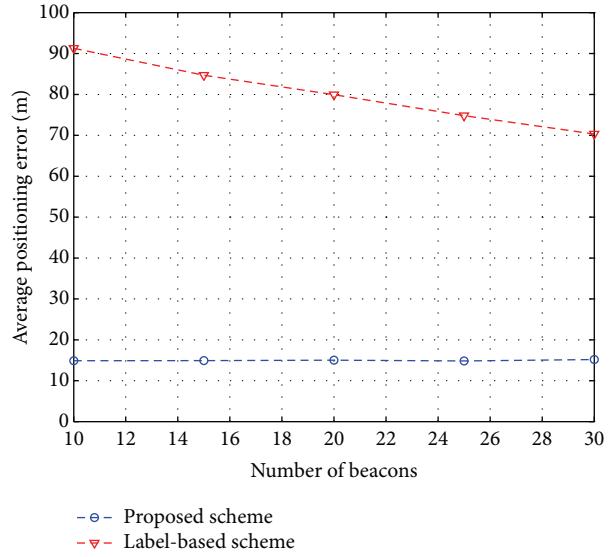


FIGURE 9: The average positioning error of our proposed scheme and the label-based scheme.

$\Delta L_P = 2\sqrt{R^2 - \Delta L_H^2}/4 - R = 149.6$ m, which can minimize the power consumption introduced by the mobile beacon. It shows that our proposed scheme can achieve a much higher detection probability than the label-based scheme. And when the number of beacon nodes equals 20, that is, there are 1.4 beacons in the transmission range of the attacker in average, the detection probability of our proposed scheme is larger than 95%. Overall, our proposed scheme outperforms the label-based scheme.

Figure 8 illustrates the wormhole detection probability of our proposed scheme with different number of wormhole attacks in which the settings of ΔL_H and ΔL_P are the same

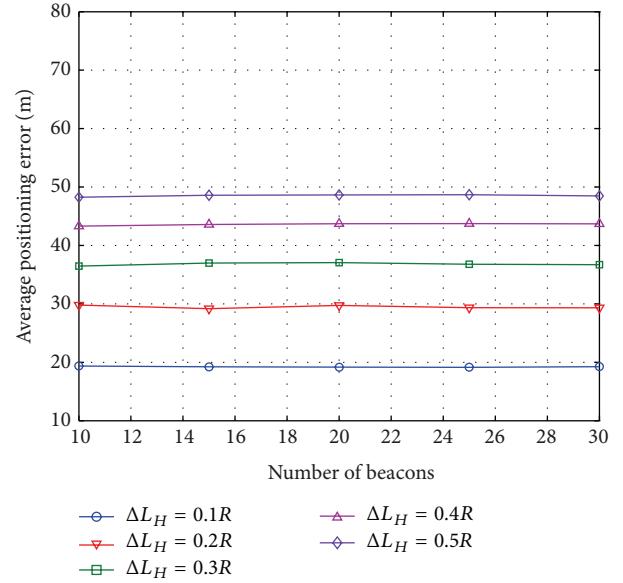


FIGURE 10: The effects of mobile beacon node's moving step length on the attacker positioning accuracy.

with that in Figure 7. It shows that when the number of wormhole attacks increases, the detection probability will decrease, which is consistent with (3). However, the detection probability is relatively high when the number of beacons is larger than 20.

Figure 9 illustrates the wormhole attacker positioning error of our proposed scheme and the label-based scheme in which the settings of ΔL_H and ΔL_P are the same with that in Figure 7. In the label-based scheme, the attacker will be estimated as the centrality of all the beacons within its transmission range. Note that the attackers positioning performance of the basic scheme and the enhanced scheme is the same; here we do not differentiate between them. It shows that our proposed scheme can achieve a much higher positioning accuracy than the label-based scheme. Furthermore, the positioning accuracy of the label-based scheme depends on the number of beacons, while our proposed scheme obtains a stable positioning accuracy with different number of beacons.

Figure 10 illustrates the effects of the moving step lengths ΔL_H and ΔL_P on the attacker positioning accuracy of our proposed scheme (here we also do not differentiate between the basic scheme and the enhanced scheme since their positioning performance is the same). We vary the value of ΔL_H from $0.1R$ to $0.5R$ with an increment of $0.1R$. And similarly, to minimize the power consumption introduced by the mobile beacon node, ΔL_P is set as the maximum value corresponding to each ΔL_H ; that is, $\Delta L_P = 2\sqrt{R^2 - \Delta L_H^2}/4 - R$. It shows that the increase of moving step length will degrade the attacker positioning accuracy. Furthermore, even when $\Delta L_P = 0.5R$, the attacker positioning error of our proposed scheme is still less than the label-based scheme.

Figure 11 illustrates the comparison of the detection times between the basic scheme and the enhanced scheme under

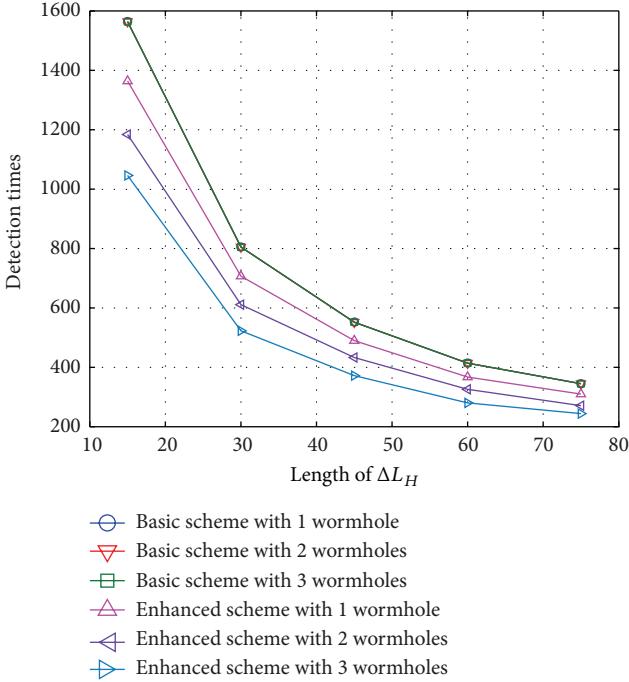


FIGURE 11: The comparison of the detection times between the basic scheme and the enhanced scheme.

different number of wormholes. The detection times here mean the total times that the mobile beacon node stop to conduct the wormhole attack detection in the whole network. Since the energy consumption in the proposed scheme mainly occurs during the wormhole attack detection, which requires the nodes to exchange messages, we can analyze the energy consumption of the scheme by directly evaluating the detection times. We adopt $\Delta L_P = 0.3R$ in Figure 11. We can observe that the number of wormholes does not affect the detection times of the basic scheme, while the increase of the number of wormholes will reduce the detection times of the enhanced scheme. Also, it shows that the proposed enhanced scheme has fewer detection times than the basic scheme, which indicates that it consumes less power than the basic scheme.

The effects of the moving step length ΔL_H and ΔL_P on the detection times of the proposed enhanced scheme are illustrated in Figure 12. The curves in Figure 12 show that the increase of moving step length, including ΔL_H and ΔL_P , can reduce the detection times; that is, it can reduce the introduced energy consumption. As the increase of the moving step length can also reduce the attackers positioning accuracy as shown in Figure 10, a tradeoff between the positioning accuracy and the energy consumption should be well balanced.

6. Conclusions

In this paper, we proposed a novel wormhole attackers detection and positioning scheme, which can not only detect the existence of wormhole attacks, but also localize the

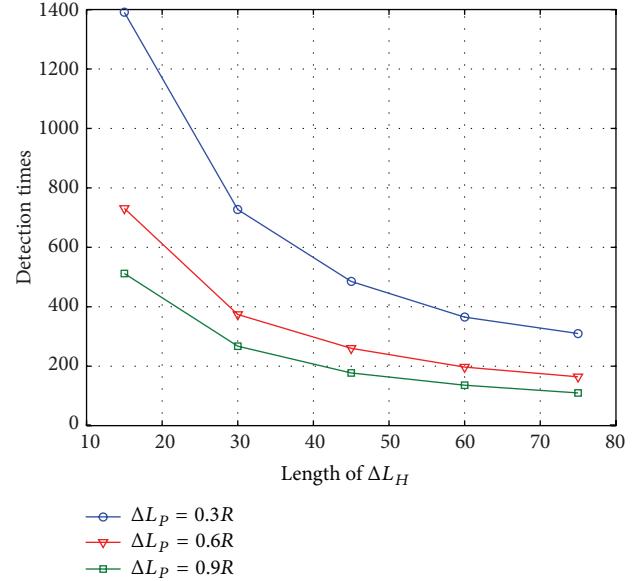


FIGURE 12: The effects of mobile beacon node's moving step length on the detection times of the proposed enhanced scheme.

attackers with a high accuracy for the system to eliminate them out of the network. The main idea is to detect whether the communication between the mobile beacon and the static beacons violates the communication properties and then the attacker can be localized as the center of its communication disk by finding the intersection point of the chords' perpendicular bisector. The simulation results illustrated that our proposed scheme can obtain a high wormhole attack detection probability together with a high accuracy for localizing the attackers.

Conflict of Interests

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

Acknowledgments

This work was supported in part by NSFC Grants no. 61309023, no. 61273079, and no. 61272463, Shandong Provincial Natural Science Foundation, China (no. ZR2013FQ032), the Fundamental Research Funds for the Central Universities (no. 13CX02100A), Open Project in Zhejiang Provincial Key Lab of Intelligent Processing Research of Visual Media (no. 2012008), State Key Laboratory of Industrial Control Technology under Grants no. ICT1206 and no. ICT1207, Hong Kong GRF Grants (PolyU-524308, PolyU-521312), and HKPU Grants (A-PL16, A-PL84).

References

- [1] S. Čapkun, “Secure positioning in wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 221–232, 2006.

- [2] J. Wu, H. Chen, W. Lou, Z. Wang, and Z. Waang, "Label-based DV-Hop localization against wormhole attacks in wireless sensor networks," in *Proceedings of the 5th IEEE International Conference on Networking, Architecture and Storage (NAS '10)*, pp. 79–88, July 2010.
- [3] Y. C. Hu, A. Perrig, and D. B. Johnson, "Wormhole attacks in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 370–379, 2006.
- [4] W. Wang, B. Bhargava, Y. Lu, and X. Wu, "Defending against wormhole attacks in mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 6, no. 4, pp. 483–503, 2006.
- [5] S. Čapkun, L. Buttyán, and J. Hubaux, "SECTOR: secure tracking of node encounters in multi-hop wireless networks," in *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor networks (in association with 10th ACM Conference on Computer and Communications Security)*, pp. 21–32, October 2003.
- [6] Y. C. Hu, A. Perrig, and D. B. Johnson, "Rushing attacks and defense in wireless Ad Hoc network routing protocols," in *Proceedings of the 2003 ACM Workshop on Wireless Security (WiSec '03)*, pp. 30–40, September 2003.
- [7] J. Eriksson, S. V. Krishnamurthy, and M. Faloutsos, "TrueLink: a practical countermeasure to the wormhole attack in wireless networks," in *Proceedings of the 14th IEEE International Conference on Network Protocols (ICNP '06)*, pp. 75–84, November 2006.
- [8] Y. Xu, G. Chen, J. Ford, and F. Makedon, "Detecting wormhole attacks in wireless sensor networks," *IFIP International Federation for Information Processing*, vol. 253, pp. 267–279, 2007.
- [9] I. Khalil, S. Bagchi, and N. B. Shroff, "LITE WORP: a lightweight countermeasure for the wormhole attack in multihop wireless networks," in *Proceedings of the International Conference on Dependable Systems and Networks (DSN '05)*, pp. 612–621, July 2005.
- [10] I. Khalil, S. Bagchi, and N. B. Shroff, "MOBIWORP: mitigation of the wormhole attack in mobile multihop wireless networks," in *Proceedings of the Securecomm and Workshops*, September 2006.
- [11] W. Wang, J. Kong, B. Bhargava, and M. Gerla, "Visualisation of Wormholes in underwater sensor networks: a distributed approach," *International Journal of Security and Networks*, vol. 3, no. 1, pp. 10–23, 2008.
- [12] R. Maheshwari, J. Gao, and S. R. Das, "Detecting wormhole attacks in wireless networks using connectivity information," in *Proceedings of the 26th IEEE International Conference on Computer Communications (IEEE INFOCOM '07)*, pp. 107–115, May 2007.
- [13] X. Ban, R. Sarkar, and J. Gao, "Local connectivity tests to identify wormholes in wireless networks," in *Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '11)*, May 2011.
- [14] D. Dong, M. Li, Y. Liu, X. Li, and X. Liao, "Topological detection on wormholes in wireless ad hoc and sensor networks," *IEEE/ACM Transactions on Networking*, vol. 19, no. 6, pp. 1787–1796, 2011.
- [15] T. Dimitriou and A. Giannetsos, "Wormholes no more? Localized Wormhole detection and prevention in wireless networks," in *Proceedings of the IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS '10)*, 2010.
- [16] H. Chen, W. Lou, and Z. Wang, "Secure localization against wormhole attacks using conflicting sets," in *Proceedings of the IEEE 29th International Performance Computing and Communications Conference (IPCCC '10)*, pp. 25–33, December 2010.
- [17] H. Chen, W. Lou, J. Wu, Z. Wang, and A. Xia, "Securing DV-Hop localization against Wormhole attacks in wireless sensor networks," *Pervasive and Mobile Computing*, 2014.
- [18] H. Chen, W. Lou, and Z. Wang, "On providing Wormhole attack resistant localization using conflicting sets," *Wireless Communications and Mobile Computing*, 2014.

Research Article

DCEP: Data Collection Strategy with the Estimated Paths in Ocean Delay Tolerant Network

Chao Liu,¹ Zhongwen Guo,¹ Feng Hong,¹ and Kaishun Wu²

¹Department of Computer Science and Technology, Ocean University of China, China

²Department of Computer Science & Engineering, Hong Kong University of Science and Technology, Hong Kong

Correspondence should be addressed to Feng Hong; hongfeng@ouc.edu.cn

Received 7 January 2014; Accepted 31 January 2014; Published 12 March 2014

Academic Editor: Lei Chen

Copyright © 2014 Chao Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Data collection is an important procedure of ocean monitoring systems. Due to limited bandwidth and extreme high cost of satellite communication, the majority of data is unable to transmit via satellites. For the moment, there are no efficient data collection methods provided for data collection in underwater sensor networks. After collecting and analyzing GPS data of 40 ships for two months, we find the mobility pattern of ships. On the basis of the mobility pattern, we propose a new data collection strategy for underwater sensor networks through delay tolerant routing. Through simulations and real data analysis, we provide quantitative analysis of the proposed strategy, such as data collection ratio, complexity of the algorithm and energy consumption. Especially, the data collection ratio on real data analysis could reach 95%–100% in acceptable time, which is 30% more than the radio of the strategy that data stored as they come.

1. Introduction

The ocean has become one of the most important treasures of people's resources with the development of human society and also has significant impact on the change of global climate. Therefore, the research of the ocean has become one of the most important development strategies of the costal countries for many years. Long-term automatic monitoring is a common technique for the study of the ocean [1–3]. Due to the limited bandwidth and the extreme high cost of satellite communication, the majority of the data is unable to be transmitted in real time. In this case, the traditional method is the local storage of the mass data and regular collection by special data mules after a long cycle interval [2–4] or using AUV as substitution for sensors [5–8]. However, these approaches not only reduce the efficiency due to difficulty in obtaining the real-time data but also significantly increase the cost.

To reduce the cost, delay tolerant network provides a by-product effective technical approach for data collection problem [9–11]. The designers use mobile devices which always travel between sources and destinations by the way, such as buses or robots, to deliver messages [9, 10, 12–14].

Motivated by this, we observe that many ships work on the ocean every day, and some of them may pass across the monitoring area. Taking advantage of these passing ships to help the data collection may be a feasible plan.

However, unlike the delay tolerant network on land, ocean delay tolerant network has different characteristics and some challenges are as follows. First, the ocean network usually adopts acoustic communications. Different from the terrestrial wireless communications, the acoustic communications have the properties of low bandwidth, high latency, and high error rate and are seriously affected by multipath transmission and Doppler effect [15–17]. The data transmission between ships and sensors is achieved by acoustic communications. Because the transmission rate of underwater sensors is very low and the traffic volume of ships is not as much as that of vehicles, increasing the average transmission efficiency is a more urgent problem of data collection. Second, the mobility patterns of ships and vehicles are different. The vehicle trace is strictly confined by roads [14, 18]. In other words, different vehicles have the same trace when they pass through the same road. However, there are no roads in the ocean like overland. We need to

find the mobility pattern in the ocean which could support our work. Through this pattern, we may generate paths with different probabilities from large amounts of GPS data. Based on the different probabilities of the paths, we propose to design a strategy to allocate data to different sensors in the monitoring area. Third, there is no routing plan designed for data collection in ocean delay tolerant network. In order to increase the transmission efficiency and the data collection radio, we need to route data to the sensors which can easily communicate with ships. Many effective routing protocols have been designed for delay tolerant network [18–22]. However, some of them are designed for message delivery which cannot be adopted for massive data transmission because of the short interactive period, and others are context-related or probability-related routing plan which cannot realize the data routing of multipoint to paths with different probability. We need to design a routing plan which could distribute the data of the whole network to several paths with different probabilities and realize massive data transmission.

In order to investigate the mobility pattern of ships, we collect the trace data of 40 ships for two months (from 2011-07-01 to 2011-08-31). From these data as shown in Figure 1, we observe that the trace of ship is not regular as that of vehicle despite of the navigational aid equipped. Through analyzing the historical GPS data, we propose a new method to find out several approximately estimated paths of a monitoring area that the ships would pass through.

In this paper, we provide an efficient data collection strategy based on the mobility pattern of ships. To the best of our knowledge, this is the first detailed, systematic data collection strategy in ocean monitoring field. And it is the first paper report on mobility pattern of ships with real data at this scale. We summarize the main contributions of this paper as follows.

- (1) We propose an innovative method to generate estimated paths by summarizing the general rule from large amounts of ship traces collected by experiments. The estimated paths could be used in data allocation and data routing procedure.
- (2) We provide a routing plan of multipoints to paths of different probabilities. We transform the data routing problem to the assignment problem. Because of the specific characteristic of this assignment problem, in order to save computing resources, we design a greedy algorithm and its complexity is $O(n^2)$.
- (3) We design a data collection strategy to collect data from the sensors based on the mobility pattern of ships. In this system, we take advantage of the mobility pattern of ship which we have summarized and the routing plan we have provided to boost the speed of data collection by routing the data to the sensors near estimated paths.
- (4) We conduct extensive trace-driven experiments, and the experimental results show that the collection ratio of our strategy could reach 95%–100% in acceptable time, which is 30%–150% more than the radio of the strategy that data stored as they come.

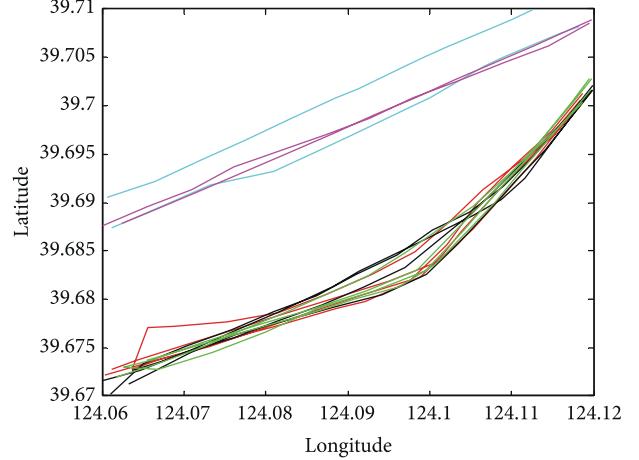


FIGURE 1: The traces of ships (from 2011-08-01 to 2011-08-06).

The paper is organized as follows. In Section 2, we introduce the characteristics of ship traces and provide a method to generate the estimated paths of ships. The design of the data allocation strategy is described in Section 3. The data routing problem is discussed in Section 4. In Section 5, we provide the simulation with both simulated data and real data. We survey the related work in Section 6. Finally, conclusions are presented and suggestions are made for future research in Section 7.

2. Estimated Path Generation

In this section, we will discuss the mobility pattern of ships. Based on the pattern, we describe the definition of the distance between two curves and the method of estimated paths generation which can be used during data collection.

2.1. Trace Analysis. As we discussed in Section 1, the traces of ships have the specialty of no road confined. In order to find the mobility pattern of ships, we put our devices on 40 ships in The Yellow Sea for two months (from 2011-07-01 to 2011-08-31) to collect their GPS data.

Through analyzing the GPS data of ships, we preliminarily find that the sailing traces of the ships are regular as Figure 1 shows. The traces of different ships of same harbor sailing to one target are all different every time; however, these traces seem to be confined by an invisible boundary. Even if the traces are not like the vehicle's which can be confined by roads in tens of meters, but the underwater sensor's large communication range could tolerate the error of hundreds of meters.

As these similar traces are regular and confined within range, they could be treated as one trace just like the overland roads. Apparently, any of them could not represent for others. So, we provide a method which could fuse several paths from these historical traces which could summarize the general rule of these traces. Transmitting data to the sensors near these fusion paths could make the data collection work

FIGURE 2: The example of distance between curves a and b .

efficient, because these paths could roughly represent the ships' sailing paths.

2.2. Distance between Two Curves. Before path fusion, we will introduce a new definition, the distance between two curves.

We assume that the distance between curves a (r_a) and b (r_b) is d_{ab} . For every point x_{ak} ($k = 1, 2, \dots, n, \dots$) of r_a , calculate the minimum distance l_{ak} ($k = 1, 2, \dots, n, \dots$) from x_{ak} to r_b . Similarly, we can work out every l_{bk} ($k = 1, 2, \dots, n, \dots$) of r_b . The distance d_{ab} between curves a and b is defined as follows:

$$d_{ab} = \max \{l_{a1}, l_{a2}, \dots, l_{an}, \dots, l_{b1}, l_{b2}, \dots, l_{bn}, \dots\}. \quad (1)$$

Figure 2 is the example of the distance between two curves. As shown in Figure 2, obviously, l_{ai} is the maximum of $\{l_{a1}, l_{a2}, \dots, l_{an}, \dots, l_{b1}, l_{b2}, \dots, l_{bn}, \dots\}$. So, l_{ai} is the distance between curves a and b .

2.3. Path Fusion. The distance between two curves can effectively express the similarity of two curves. The shorter it is, the more similar they are. We can use this definition to support our path fusion (Algorithm 1).

For each path trajectory across the monitoring area, we design a structure that contains four contents, the *UID*, the *GPS_data*, the *nearPathIds*, and the *isFusion*. The *UID* contains the unique identifier of one path. It is the combination of the timeline and the ship's *UID*. For example, a ship whose *UID* is 016, it entered the area at 2012-07-05 02:35:06. The path's *UID* is 20120705023506016. This *UID* is surely unique. The *GPS_data* contains the GPS data collected by the equipment on the ship. The *nearPathIds* contains an array of *UID* of the traces whose distance to this trace is less than the sensor's communication radius. The *isFusion* is a Boolean variable, which expresses whether the path is fusion by other paths. The value true stands for yes and false for no (default false).

For example, we assume a $10 \text{ km} \times 10 \text{ km}$ area as shown in Figure 3 and the communication radius is 1 km.

UID of path 20110706023506016 is 20110706023506016. Because the distance between path 20110706023506016 and 20110706023594017 is less than 1km, its *nearPathIds* is 20110706023594017. As it is not fused by other paths, its *isFusion* is false.

Path 2011070623045016 has its *UID* 2011070623045016. Because there are no paths whose distance is less than 1km with this path, its *nearPathIds* should be empty. As it is not fused by other paths, its *isFusion* is false.

After generating a new path, it should be judged whether it is near all the other paths generated before. If so, the new

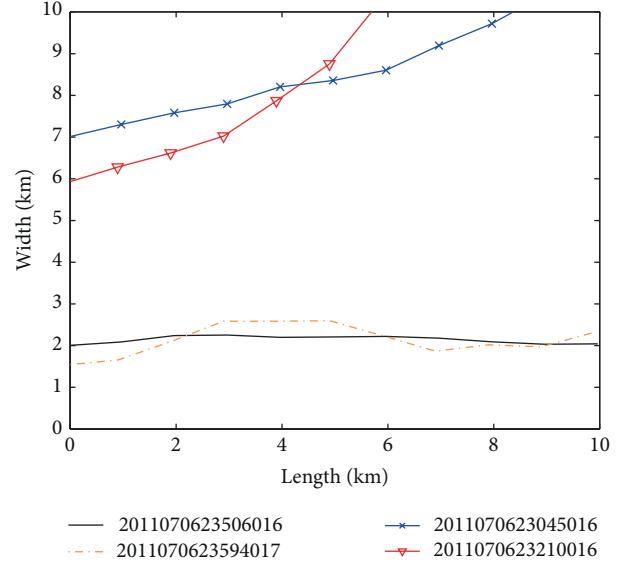


FIGURE 3: Example of traces.

path and the paths near to it should update their *nearPathIds* array.

The pathFusion algorithm will do the initialization work first. We should initialize set S which stores the path waiting to be fused in one fusion cycle and set F which stores the fusion path fused in S .

The function *sortDesc* deals with the sorting procedure. The paths should be sorted by the length of their *nearPathIds* from maximum to minimum. This procedure is designed to avoid generating too many fusion paths that reduce the efficiency of data allocation.

After selecting one path, the function *nearPathSet* is used to generate a set contains the paths whose *UID* is in the *nearPathIds* array of this path and its *isFusion* is false. Use function union to union the set generated by *nearPathSet* and this path and mark the elements' *isFusion* to be true.

The function *traceFusion* deals with the main process of path fusion. For every path in S , evenly discretize these paths into M (M is big enough) pieces from the same side in this area (because paths may have different directions) and number the points sequentially from 1 to M . For the points of same number, work out, respectively, the mean number of longitude and latitude. After that we would have got M new points' coordinates numbered from 1 to M . Link these points in sequence and assign the number of elements in S to the weight factor of this fusion path (the larger the weight factor is, the more probable ships pass this trace). Finally, put the fusion path into F .

3. Data Allocation Strategy

To increase the efficiency of data collection, we need to route the data to the sensors near the paths through which ships always pass. After summarizing the mobility pattern of ships, we know that the fusion paths could approximately represent the general rule of the traces. In this section, we

```

Input: Array of paths  $path[N]$ ;
Output: Fusion path set  $F$ ;
sortDesc( $path$ ,  $path \cdot nearPathIds \cdot length$ );
for each  $path[i]$  do
     $path[i] \cdot isFusion = false$ ;
End
for each  $path[i]$  do
    if  $path[i] \cdot isFusion == false$  then
         $S = \text{union}(path[i], \text{nearPathSet}(path[i]))$ ;
        Mark elements'  $isFusion$  in  $S$  to be true;
        if  $S \cdot length > 1$  then
             $fPath \cdot trace = \text{traceFusion}(S)$ ;
             $fPath \cdot weight = S \cdot length$ ;
        else
             $fPath \cdot trace = path[i] \cdot GPS\_data$ ;
             $fPath \cdot weight = 1$ ;
        end
        Put  $fPath$  into  $F$ ;
        Clear  $S$ ;
    end
end
return  $F$ ;

```

ALGORITHM 1: Path fusion.

present a data allocation strategy based on the fusion paths calculated before. All the derivation processes are in these two assumptions: (1) all ships pass through the area in same speed v and (2) all the sensors have the same communication radius r and same transmission rate b .

3.1. Transmission Capacity of Path. Transmission capacity is the ability of data collection of one path. It is an important factor which determines the allocation strategy. In this part, we will calculate the transmission capacity of a path.

Assume we select a fusion path $path_i$. It is easy to calculate the distance d_i that the ship passing every sensor node in the region of communication. Assume sailing along the path could communicate with n sensors, the transmission capacity of path could be calculated as follows:

$$c = \frac{\sum_{i=1}^n d_i}{v} \times b. \quad (2)$$

3.2. Data Allocation Strategy in Path Level. Based on the weight factor of fusion paths, we cannot directly find out how much data should be stored in a sensor. However, taking advantage of the weight factor and the transmission capacity of each fusion path, we can calculate how much data should be allocated, respectively, to these paths.

Assume that a ship that passes through the area along the different fusion paths could take the same amount of data A , and total amount of data of all sensor nodes in the area is M . To collect all the data in this area, we need ships to cross the area by at least $m = M/A$ times.

Through the weight factor, it is easy to calculate the probability of each path that the ships would pass through.

Assume we have fused n_r fusion paths and the probability p_i of each path. It is easily known that $\sum_{i=1}^{n_r} p_i = 1$.

For one fusion path, we could know that a ship has probability p_j along path j to cross the area and the probability $1-p_j$ along others. Thus, the probability of ship passing through path j obeys binomial distribution

$$p(x=k) = \binom{m}{k} p_j^k (1-p_j)^{m-k}, \quad k = (0, 1, 2, \dots, m), \quad (3)$$

where m is M/A .

Because it obeys binomial distribution, the most probability of crossing times along path j is $\lfloor (m+1)p_j \rfloor$. Similarly, for every fusion path, the most probability of crossing times is $\lfloor (m+1)p_i \rfloor$, $i = (1, 2, 3, \dots, n_r)$.

Obviously, if we distribute the data proportionally by p_i , it is most probable that all the data could be collected after m ships passing through the area.

However, the reality cannot be ideal as what we discussed above. Please imagine such a situation. A ship has 50% probability passing through this area along path i and 50% along path j , but it could collect data of 100 MB passing along path j and only 1 MB along path i . In this case, distributing data by probability could not collect the data efficiently.

Assume we have fused n_r fusion paths and the probability p_i of each path ($\sum_{i=1}^{n_r} p_i = 1$). For each path, they have their transmission capacity c_i and the amount of data m_i distributed along every path ($\sum_{i=1}^{n_r} m_i = M$).

To collect the data of a single path, it needs $\lceil m_i/c_i \rceil$ ships to pass along. As what we discussed before, the most probable

number of ships passing through the whole monitoring area k_i should be expressed as (4)

$$k_i = \left\lceil \frac{[m_i/c_i]}{p_i} \right\rceil - 1. \quad (4)$$

On the analogy of Cannikin law, to collect all the data, we should find the maximum of k_i in the kind of distribution strategy. We obtain

$$k_{\max} = \max \{k_1, k_2, \dots, k_{n_r}\}. \quad (5)$$

For every kind of data distribution strategy, we could get a k_{\max_n} , ($n = 1, 2, \dots, \infty$). We need to find the strategy that could get the minimum of k_{\max_n} . Easily, we obtain

$$k_{\min} = \min \{k_{\max_1}, k_{\max_2}, \dots, k_{\max_n}\}. \quad (6)$$

We remove the ceil part in order to make the calculation easy to derive. From (4), it is obvious that $m_i = c_i p_i (k_i + 1)$. So, the total amount of data M could be expressed as follows:

$$\begin{aligned} \sum_{i=1}^{n_r} m_i &= c_1 p_1 (k_1 + 1) + c_2 p_2 (k_2 + 1) + \dots \\ &\quad + c_{n_r} p_{n_r} (k_{n_r} + 1) = M. \end{aligned} \quad (7)$$

Now, we prove that, when $k_1 = k_2 = \dots = k_{n_r}$, we have the $k_{\min} = k_1 = k_2 = \dots = k_{n_r}$.

Assume there is a $k_x < k_{\min}$. Because $c_i p_i$ and M are constants. From (7), we could know that, if $k_x < k_{\min}$, there must be a $k_y > k_{\min}$, ($x \neq y$). So, $k_{\max_n} \geq k_y > k_{\min}$.

According to the proof process, our assumption is right. We obtain

$$\begin{aligned} \sum_{i=1}^{n_r} m_i &= c_1 p_1 (k_{\min} + 1) + c_2 p_2 (k_{\min} + 1) + \dots \\ &\quad + c_{n_r} p_{n_r} (k_{\min} + 1) = M. \end{aligned} \quad (8)$$

Finally, we get the strategy which is most probable to collect the data using the minimum number of ships. The data amount of every path is calculated as (9)

$$\begin{aligned} m_i &= M \times \frac{c_i p_i (k_{\min} + 1)}{\sum_{i=1}^{n_r} c_i p_i (k_{\min} + 1)} \\ &= M \times \frac{c_i p_i}{\sum_{i=1}^{n_r} c_i p_i}. \end{aligned} \quad (9)$$

3.3. Data Allocation Strategy in Node Level. From Section 3.2, we can get the data allocation strategy in path level. For a path which has data m , also, we assume sailing along the path could communicate with n sensor and the communication distance d_i of each sensor.

Obviously, on the condition of assumptions (1) and (2), the transmission capacity of each sensor is proportional to the sensors' communication distance with this path. Let storage_i

be the amount of data that should be stored in the node. The storage_i should be expressed as:

$$\text{storage}_i = m \times \frac{d_i}{\sum_{i=1}^n d_i}. \quad (10)$$

So far, the allocation procedure is finished. We have the data amount that every sensor should store. The next question is how to route the data of whole network to these sensors.

4. Data Routing

In previous section, we have proposed the data allocation strategy of the monitoring area. From the strategy, we could know the ideal data volume of each sensor. In this case, some sensors need to send data, and others need to receive data to change the distribution of the data in the area in order to accord with the strategy's result.

The network's life time is an essential problem of wireless sensor networks [15, 17, 23, 24]. Minimizing energy consumption during data routing process is the task that needs to be solved in this section. The energy cost that a packet routes from one sensor to different target sensors is commonly different. So, selecting the targets of different packets to minimize the whole networks energy consumption is the core problem in this section.

4.1. The Process of Data Routing. Before discussing, we need to make two hypotheses: (1) any two sensors in the monitoring area could communicate with each other by multihop and (2) the locations of all the sensors have already been known.

The whole process of data routing is shown in Figure 4. To work out the routing plan, lots of preparations need to be finished. As shown in Figure 4, we should first get the data allocation strategy and the shortest paths between every two sensors. Through the method we have described in previous section, we could get the data allocation strategy based on fusion paths. Because the location of the sensors and the communication radius of the sensor have already been known, we could calculate the adjacency matrix based on the locations. Use Dijkstra algorithm to find out the shortest paths between every two sensors [25]. These paths contribute to not only the data routing but also the calculation of the energy consumption of packet routing. With the help of these shortest paths, we could easily work out the minimum energy cost that a packet routes between every two sensors. Through the data allocation strategy and the energy cost between every two sensors, we could use our method which would be discussed in next subsection to find the optimal routing plan. The last step of this procedure is routing the data as the plan and waiting for the ships to collect the data.

4.2. The Optimal Routing Plan. Based on the strategy we discussed in Section 3, we have got the data amount S'_i that every sensor needs to store. Also, we could easily know the data amount S_i that the sensor originally has. The minimum energy consumption which a packet routes from one sensor to another is a constant that could be calculated based on the shortest routing path. If $S'_i > S_i$, it means the sensor needs to

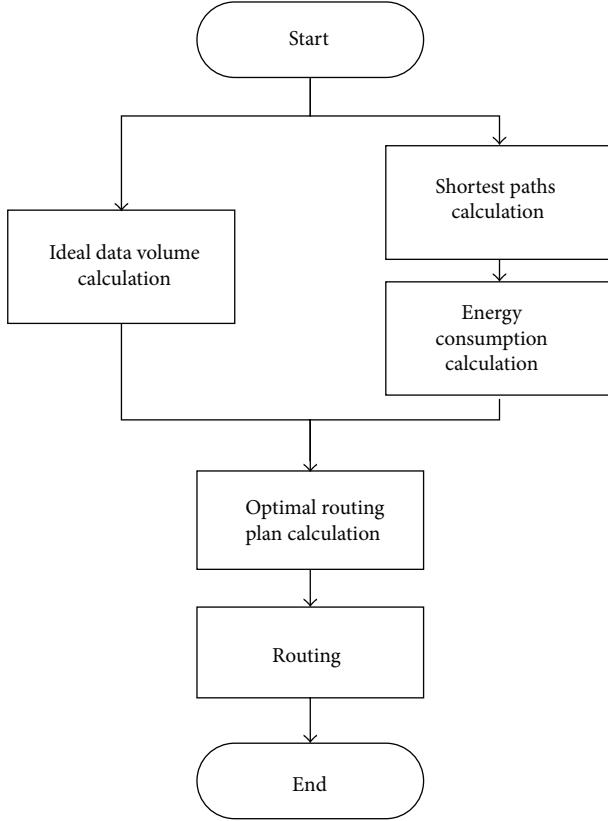


FIGURE 4: Process of data routing.

receive $(S'_i - S_i)$ packets from source sensors. Conversely, the sensor needs to send $(S_i - S'_i)$ packets to target sensors.

Assume there are N packets waiting to be sent. Also, there must be N targets waiting receiving. For each sending packet, there are N targets and N energy consumption values corresponding to each target. If packet i chooses target j to send, target j should not receive other packets. That means that the packets and the targets are one-to-one correspondence. Now, we define a matrix E of size N

$$E = [e_{ij}]_{N \times N}, \quad (11)$$

where e_{ij} means that packet i routing to target j would cost at least e_{ij} energy.

For example, there are three sensors which could communicate with each other. We assume S_a , S_b , and S_c are 4. After allocation, S'_a and S'_b should be 6. S'_c should be 0. So, sensors a and b should, respectively, receive 2 packets. Sensor c should send 4 packets to others. A packet routing from sensor a to b would cost 2 units of energy. And b to c would cost 1 unit of energy. And c to a should cost 3 units of energy. We have 4 packets in sensor c waiting to be sent and 4 targets in sensors a and b . We number these 4 packets from 1 to 4. Targets in sensor a would be numbered 1 and 2, and targets in sensor b

would be numbered 3 and 4. So, the matrix E of size 4 would be represented as follows:

$$E = \begin{pmatrix} 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 \end{pmatrix}. \quad (12)$$

For energy consumption matrix E_N , in order to find the minimum energy consumption routing plan, we need to find the minimum sum of N elements in different rows and columns. The selected row numbers corresponding to columns numbers are the final routing plan.

We assume that optimal plan $R(E_N) = \{e_{1j_1}, e_{2j_2}, \dots, e_{Nj_N}\}$. So, the minimum energy consumption of the routing plan is $\sum_{i=1}^N e_{ij_i}$. Element e_{ij_i} means the energy cost that packet i routes to target j_i and packet i should route to j_i . For example, we have matrix E as follows:

$$E = \begin{pmatrix} 1 & 2 & 1 \\ 1 & 3 & 7 \\ 1 & 5 & 4 \end{pmatrix}. \quad (13)$$

Through calculation, we could obtain that

$$R(E) = \{e_{13}, e_{22}, e_{31}\}. \quad (14)$$

The minimum energy consumption is $e_{13} + e_{22} + e_{31}$. That means packet 1 routes to target 3, packet 2 routes to target 2, and packet 3 routes to target 1. We could traverse all the combination to find the optimal combination of the elements. Its time complexity is $O(n!)$. However, if N is too large, it may cost several days to work out the answer. Fortunately, this matrix problem could be equivalent to assignment problem. It has been solved by Kuhn in 1955 using the method named Hungarian algorithm which could achieve a $O(n^3)$ running time [26]. However, if large amounts of packets need to be transmitted, this algorithm will cost several hours to find the optimal transmission plan.

4.3. The Greedy Algorithm. A greedy algorithm is a heuristic algorithm which makes the optimal choice at each procedure, hoping to find the global optimum. Although the greedy algorithm cannot find the global optimum in many problems, it could find an approximate global solution in a reasonable running time (Algorithm 2).

Our algorithm is very simple. Start from the first row, search the minimum elements in this row, and remove this column from the matrix. Go ahead for the next row until the last one. This method has $O(n^2)$ running time. Normally, the result of this method would be far greater than the optimal one in a random matrix. However, in this context, many packets are in the same sensor and so are the target spaces. That means the energy matrix has so many elements with same value. This would be the reason that makes our algorithm effective. We verified this method in the process of simulation. The result shows that the result of the greedy algorithm is a little greater than the optimal one, but the running time of this algorithm is far less than that of the optimal one. If the system lacks computing ability, we could choose this method to have an approximate solution.

```

Input: Energy consumption matrix  $E_N$ ;
Output: The routing plan routingPlan;
          Approximate result set  $R(E)$ ;
For  $i = 1$  to  $N$  do
     $[e_{ij}, j] = \min(E_N(i,:))$ ;
    Put  $e_{ij}$  into  $R(E)$ ;
    Let  $E_N(:, j) = \text{INF}$ ;
    Put  $[i, j]$  into routingPlan;
end
return  $R(E)$ , routingPlan;

```

ALGORITHM 2: Greedy algorithm.

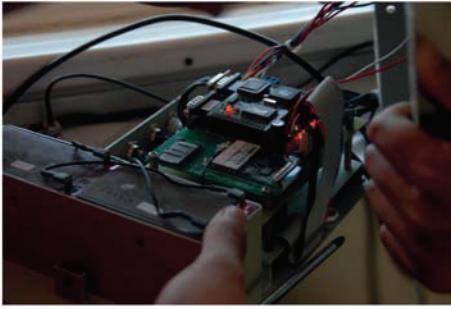


FIGURE 5: The GPS data collection devices.

5. Performance Evaluation

In this section, we introduce GPS data collection work and the simulation environment and present the result in detail. We use both simulated data and real data of ships to verify our strategy. We assume that the communication radius of the sensors and ships is 1 km. The transmission rate is 10 KB/s. The data stored in each sensor are 5 MB. The ships' speed is 15 knots. In our simulations, three metrics are evaluated and defined as follows.

Data Collection Ratio (DCR). DCR is defined as the proportion of the packets successfully collected by ships.

Energy Cost Ratio (ECR). ECR is defined as the energy cost proportion that the greedy algorithm compares with the Hungarian one, which is a key metrics to make comparison between the Hungarian algorithm and our greedy algorithm.

Running Time Ratio (RTR). RTR is defined as the running time proportion that the greedy algorithm compares with the Hungarian one during the process of finding optimal solution of data routing.

In every scenario, we would contrast our strategy to the condition that all the data are not allocated and wait the ships to pass through for collection. We call the strategy that data wait for collecting original strategy (OS) and our strategy DCEP.

5.1. Ship Trace Collection. In order to find the mobility pattern of ships, we put the devices (as Figure 5 shows) with general packet radio service (GPRS) modules DTP-S09 on ships. Each device is equipped with a SIM card. The GPS data is transmitted through the VPN of National Bureau of Oceanography provided by China Mobile Communications Corporation (CMCC) to our database every minute. Through two months (from 2011-07-01 to 2011-08-31) work, we collect the GPS data of 40 ships.

5.2. Simulation. In first scenario, we set a $10\text{ km} \times 10\text{ km}$ area and 3 fusion paths that their weight factors are, respectively, 3, 2, and 1. Ships crossing the area would select the path by their weight factors. That means it has the 50% probability of sailing along path 1, 33.3% along path 2, and 16.7% along path 3. To be closer to real life, the sailing paths would not be strictly the same with the fusion path. We add some random Gaussian distribution disturbances when the ship sails along one fusion path. We run our simulation in two different conditions, sensors equally deployed and sensors randomly deployed.

Figure 6(a) illustrates the result on data collection ratio of the condition that 169 sensors are equally deployed in the area. As more ships cross the area, the DCR is continuously increasing in DCEP. Comparing DCEP and OS, with increasing amount of ships passing through the area, the DCR of DCEP continues to rise until 94.80%. DCR of OS raises nearly the same with DCEP before the passing of 5 ships; however, the growth of the DCR starts to decline after the passing of 5 ships and DCR finally stops at 35.71%.

Figure 6(b) illustrates the result on data collection ratio of the condition that 220 sensors are randomly deployed in the area. All results present the same rule as the sensors are equally deployed. With increasing amount of ships passing through the area, the DCR of DCEP continues to rise until 97.91%. However, the growth of the DCR starts to decline after the passing of 5 ships and DCR finally stops at 36.33%.

5.3. Real Data Analysis. As we mentioned in Sections 1 and 2, we collect the GPS data of 40 ships in The Yellow Sea of China over two months. In the second scenario, we use the real data of ship traces to run our simulation. In the process of data analysis, we found that different ships used similar path

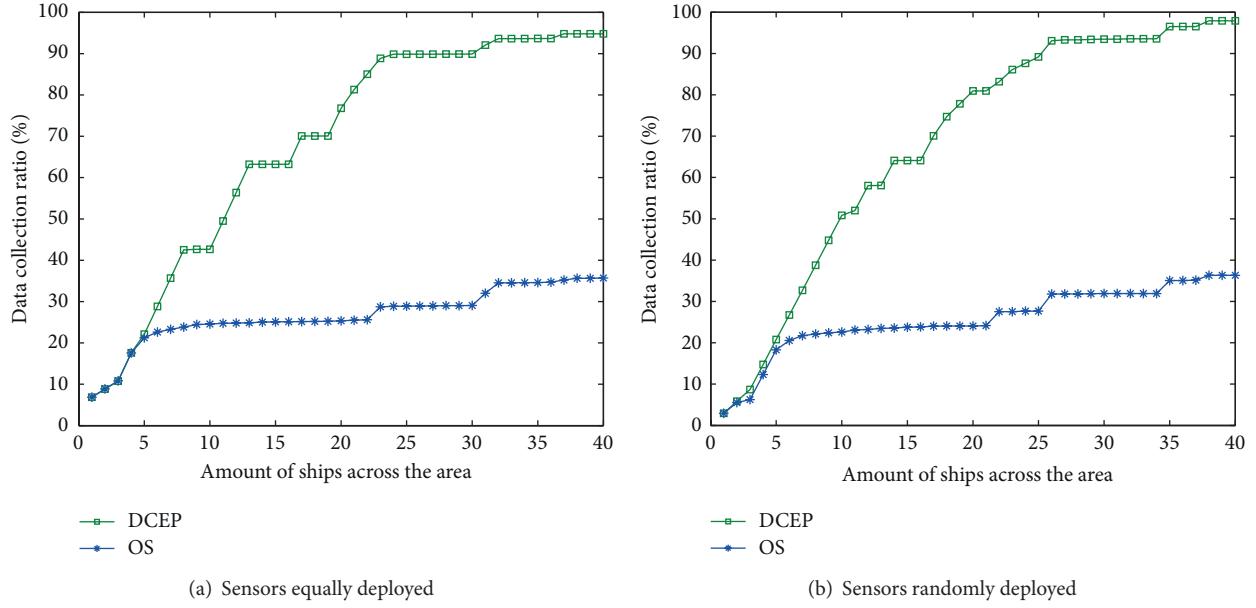


FIGURE 6: Data collection ratio of simulated data.

while sailing from one certain location to a certain target. So, we choose a region from 124.06°E to 124.12°E and 39.665°N to 39.71°N to verify our strategy. As shown in Figure 8, we use the GPS data from 2011-08-01 to 2011-08-05 (color red) to generate two estimated paths (color blue). We use the traces in the next several days to simulate the procedure of data collection.

Figure 7(a) illustrates the result on data collection ratio of the condition that 49 sensors are equally deployed in the area. With increasing amount of ships passing through the area, the DCR of DCEP continues to rise until 100.00%. DCR of OS raises nearly the same with DCEP at the beginning. However, the growth of the DCR starts to decline after the passing of 4 ships and DCR finally stops at 79.92%.

From Figure 7(a), we find that the OS collects more data than the DCEP at the beginning. Some sensors away from the fusion paths may store little or even no data after the allocation procedure. However, all the sensors have data at the beginning in OS. Because the ships are not strictly stucked to the fusion path during sailing, the DCEP may collect less data than OS at the beginning due to passing along sensors away from fusion paths, but after several ships passed, the DCR of DCEP would increase faster.

Figure 7(b) illustrates the result on data collection ratio of the condition that 50 sensors are randomly deployed in the area. With increasing amount of ships passing through the area, the DCR of DCEP continues to rise until 100%. However, the growth of OS DCR starts to decline after 4 ships passing and the DCR finally stops at 75.38%.

5.4. Algorithm Evaluation. During the process of simulation, we evaluate the performance of two algorithms during data routing process. From Table 1, we find that the energy cost of the greedy algorithm is nearly 11% more than that of the Hungarian one. However, the running time of the greedy

TABLE 1: ECR and RTR.

Results	Simulated data work		Real data work	
	Equally	Randomly	Equally	Randomly
ECR	111.48%	111.15%	109.73%	111.81%
RTR	0.48%	0.32%	2.3%	2.1%

algorithm is only a small part of the Hungarian one's. The data capacity of first scenario is 800–1100 MB, and data capacity of first scenario is 200–300 MB. As our result shows, the more data are stored in monitoring area, the more efficient the greedy algorithm will be without wasting more energy.

6. Related Work

In this section, we introduce some researches related to our paper. The research on underwater environment monitoring has drawn amount of attention, and some real-world systems have been deployed [1, 3, 5, 6]. Also some researchers have designed some strategy to monitor the ocean [4]. There are three kinds of data collection strategy in these systems. The first strategy of data collection is to deploy wired network to realize the data transmission. SNUSE is deployed for seismic monitoring and oil exploitation. The underwater network uses the wired sensors to realize collaboration of each sensor [3]. This kind of system could realize the effective and real-time data collection. However, building and maintaining a wired underwater network would use large costs. The second strategy is to collect data through satellites. Sea-web introduces a US Navy's underwater sensor network which can monitor the condition of underwater. The sensor reports its data to the buoy which could communicate with the base station by satellite or radio [1]. But satellite communication would not only increase the cost but also shorten the lifetime

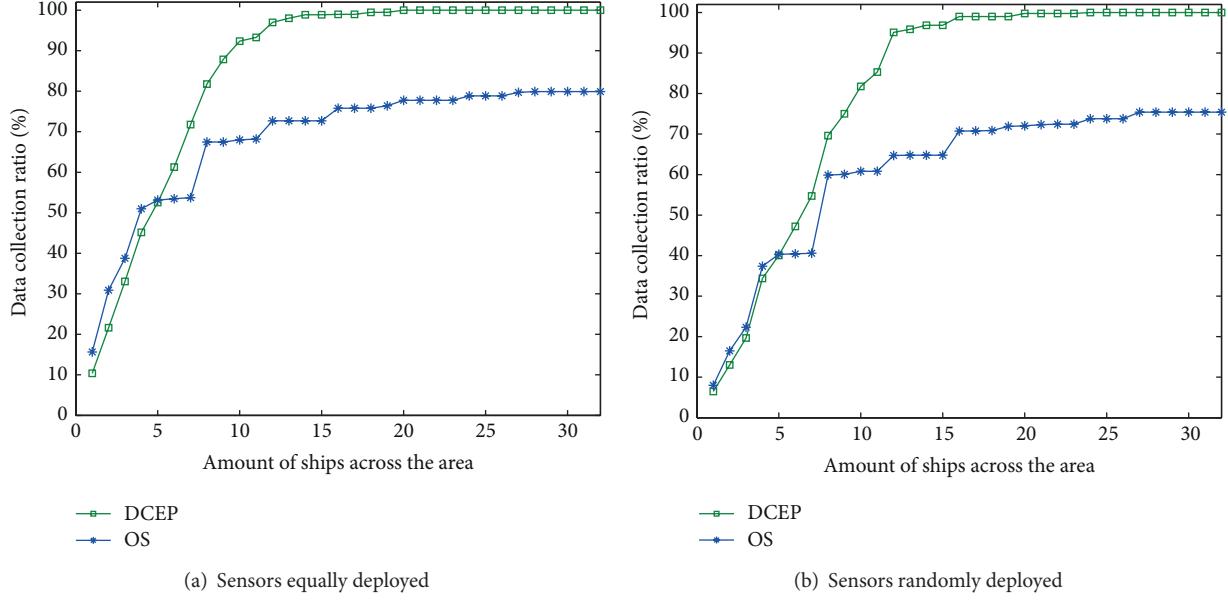


FIGURE 7: Data collection ratio of real trace data.

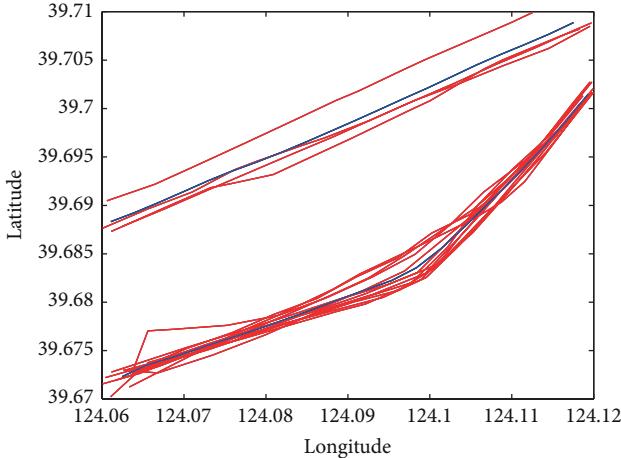


FIGURE 8: Fusion path and traces from 2011-08-01 to 2011-08-05.

of the network. The third strategy is using data mules to realize environment monitoring or data collection. The AOSN and ASAP are monitoring schemas deploying autonomous underwater vehicles (AUVs) to monitor the area and predict the future conditions. They collect the data through glider network, base station, and the recovery of AUVs [2, 5]. This kind of strategy needs to deploy special equipments for long-term work. Our team did lots of researches on environment monitoring [27–29].

The mentioned researches are all designed for underwater environment monitoring. To achieve data collection, the systems always use satellites, data mules, and wired sensors, which significantly increase the cost. The delay tolerant network provides a by-product method to realize data collection or message delivery. In [30], Zhang et al. describe a delay tolerant network to monitor the behavior custom of

wild animal called ZebraNet. DakNet wireless network takes advantages of the transportation which has communication modules to distribute the connectivity to developing villages [10]. Burgess et al. set an experiment which use buses to test his routing algorithm [18]. The above systems use the concept of delay tolerant network to collect or transmit the data by making use of the mobility pattern of animals and vehicles. For further study on mobility pattern of vehicles, Zhu and Ni present an innovative scheme to collect vehicles GPS data [31] and summarize the intercontact time and mobility pattern in urban VANETs [32]. He et al. provided some target tracking strategies with WSN [33, 34]. Li et al. designed an algorithm of sensor network navigation without locations [35]. However, the prior researches focus on the terrestrial network.

In this paper, we conduct an experiment to collect the GPS data of 40 ships to discover the mobility pattern of ships. Based on the mobility pattern of ship, we provided a data collection strategy which uses the ships of daily work to achieve data collection without the help of special data mules.

7. Conclusion

In this paper, we present a data collection strategy based on estimated path in ocean delay tolerant network. Through performance evaluation with both real data and simulated data, we prove that our method is effective. By this strategy, the data monitoring system could collect the data faster than ever without the help of special data mules. We also provide a feasible method to summarize the mobility pattern of ships which could be used to further study of the intrusion detections and location forecast.

Nevertheless, many issues still remain to be explored. Our ongoing works are (1) finding out a plan that solves this problem under the circumstance that different ships have

different speed and (2) providing a strategy that could solve this problem under sparse sensor deployment.

Conflict of Interests

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

Acknowledgment

This work was supported by the National Natural Science Foundation of China Grants 61170258, 61103196, 61379127, and 61379128.

References

- [1] J. Rice and D. Green, "Underwater acoustic communications and networks for the US navy's seaweb program," in *Proceedings of the 2nd International Conference on Sensor Technologies and Applications (SENSORCOMM '08)*, pp. 715–722, Cap Esterel, France, August 2008.
- [2] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data collection, storage, and retrieval with an underwater sensor network," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 154–165, 2005.
- [3] SNUSE, <http://www.isi.edu/ilense/snuse/>.
- [4] M. Li, Z. Yang, and Y. Liu, "Sea depth measurement with restricted floating sensors," *ACM Transactions on Embedded Computing Systems*, vol. 13, no. 1, article 1, 2013.
- [5] Autonomous Ocean Sampling Networks II, <http://www.princeton.edu/~dcs1/aosn/>.
- [6] ASAP, <http://www.princeton.edu/~dcs1/asap/>.
- [7] V. Bokser, C. Oberg, G. Sukhatme, and A. Requicha, "A small submarine robot for experiments in underwater sensor networks," in *Proceedings of the International Federation of Automatic Control Symposium on Intelligent Autonomous Vehicles (IFAC '04)*, 2004.
- [8] M. Dunbabin, J. Roberts, K. Usher, G. Winstanley, and P. Corke, "A hybrid AUV design for shallow water reef navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '05)*, pp. 2105–2110, Barcelona, Spain, April 2005.
- [9] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile Ad Hoc networks," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MoBiHoc '04)*, pp. 187–198, May 2004.
- [10] A. Pentland, R. Fletcher, and A. Hasson, "Daknet: rethinking connectivity in developing nations," *Computer*, vol. 37, no. 1, pp. 78–83, 2004.
- [11] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting mobility for energy efficient data collection in wireless sensor networks," *Mobile Networks and Applications*, vol. 11, no. 3, pp. 327–339, 2006.
- [12] A. Kansal, A. A. Somasundara, D. D. Jea, M. B. Srivastava, and D. Estrin, "Intelligent fluid infrastructure for embedded networks," in *Proceedings of the 2nd International Conference on Mobile Systems, Applications and Services (MobiSys '04)*, pp. 111–124, 2004.
- [13] A. Parker, A. Kansal, A. A. Somasundara, D. D. Jea, M. B. Srivastava, and D. Estrin, UCLA DTN Sensor Networks Update, 2004, <http://www.dtnrg.org>.
- [14] X. Zhang, J. K. Kurose, B. N. Levine, D. Towsley, and H. Zhang, "Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing," in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '07)*, pp. 195–206, September 2007.
- [15] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, "Research challenges and applications for underwater sensor networking," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '06)*, pp. 228–235, April 2006.
- [16] A. Caruso, F. Paparella, L. F. M. Vieira, M. Erol, and M. Gerla, "The meandering current mobility model and its impact on underwater mobile sensor networks," in *Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM '08)*, pp. 771–779, Phoenix, Ariz, USA, April 2008.
- [17] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [18] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: routing for vehicle-based disruption-tolerant networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–11, Barcelona, Spain, April 2006.
- [19] D. Pompili, T. Melodia, and I. F. Akyildiz, "Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MobiCom '06)*, pp. 298–309, September 2006.
- [20] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '04)*, pp. 145–158, September 2004.
- [21] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 606–620, 2007.
- [22] B. Burns, O. Brock, and B. N. Levine, "MV routing and capacity building in disruption tolerant networks," in *Proceedings of the IEEE INFOCOM 2005*, pp. 398–408, Miami, Fla, USA, March 2005.
- [23] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 3, pp. 1567–1576, June 2002.
- [24] Y. Liu and M. Li, "Iso-Map: energy-efficient contour mapping in wireless sensor networks," in *Proceedings of the 27th International Conference on Distributed Computing Systems (ICDCS '07)*, Toronto, Canada, June 2007.
- [25] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [26] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics*, vol. 52, no. 1, pp. 7–21, 2005.
- [27] Z. Guo, C. Liu, Y. Feng, and F. Hong, "CCSA: a cloud computing service architecture for Sensor Networks," in *Proceedings of the International Conference on Cloud and Service Computing (CSC '12)*, pp. 25–31, Shanghai, China, November 2012.

- [28] C. Liu, Q. Yu, T. Zhang, and Z. Guo, "Component-based cloud computing service architecture for measurement system," in *Proceedings of the IEEE International Conference on Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing and Internet of Things (iThings/CPSCom '13)*, pp. 1650–1655, Beijing, China, August 2013.
- [29] Z. Guo, C. Liu, X. Wang et al., "CVIS: complex virtual instruments system architecture," in *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC '13)*, pp. 566–571, Minneapolis, Minn, USA, May 2013.
- [30] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, "Hardware design experiences in ZebraNet," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 227–238, ACM, November 2004.
- [31] H. Zhu and L. M. Ni, "HERO: online real-time vehicle tracking in Shanghai," in *Proceedings of the IEEE 27th Conference on Computer Communications (INFOCOM '08)*, pp. 942–950, Phoenix, Ariz, USA, April 2008.
- [32] H. Zhu, L. Fu, G. Xue, Y. Zhu, M. Li, and L. M. Ni, "Recognizing exponential inter-contact time in VANETs," in *Proceedings of the IEEE INFOCOM 2010*, pp. 1–5, San Diego, Calif, USA, March 2010.
- [33] T. He, P. Vicaire, T. Yant et al., "Achieving real-time target tracking using wireless sensor networks," in *Proceedings of the 12th IEEE Real-Time and Embedded Technology and Applications Symposium*, pp. 37–48, April 2006.
- [34] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. F. Abdelzaher, "Range-free localization and its impact on large scale sensor networks," *ACM Transactions on Embedded Computing Systems*, vol. 4, no. 4, pp. 877–906, 2005.
- [35] M. Li, Y. Liu, J. Wang, and Z. Yang, "Sensor network navigation without locations," in *Proceedings of the IEEE INFOCOM 2009*, pp. 2419–2427, Rio de Janeiro, Brazil, April 2009.

Research Article

Mobile Tracking Based on Support Vector Regressors Ensemble and Game Theory

Fanzi Zeng, Shaoyuan Liu, Renfa Li, and Qingguang Zeng

Key Laboratory for Embedded and Network Computing of Hunan Province, School of Information Science and Engineering, Hunan University, Changsha 410082, China

Correspondence should be addressed to Fanzi Zeng; zengfanzi@126.com

Received 19 November 2013; Accepted 13 January 2014; Published 11 March 2014

Academic Editor: Feng Hong

Copyright © 2014 Fanzi Zeng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A two-step tracking strategy is proposed to mitigate the adverse effect of non-line-of-sight (NLOS) propagation to the mobile node tracking. This strategy firstly uses support vector regressors ensemble (SVRM) to establish the mapping of node position to radio parameters by supervising learning. Then by modelling the noise as the adversary of position estimator, a game between position estimator and noise is constructed. After that the position estimation from SVRM is smoothed by game theory. Simulations show that the proposed strategy results in the more accurate performance, especially in the harsh environment.

1. Introduction

The location service is foundation of pervasive computing, intelligent transportation, and application of WSN, so the mobile location technologies have drawn a lot of attention [1, 2]. Existing range-based location techniques include ToA, TDoA, and RSSI. For range-based location techniques, if the line-of-sight (LOS) propagation exists between the mobile node and anchor nodes, high location accuracy can usually be achieved using the conventional location algorithms [3]. However, since the direct path from the mobile node to anchor nodes can be blocked by buildings and other obstacles, the transmitted signal could only reach the receiver through reflected, diffracted, or scattered paths called non-line-of-sight (NLOS) paths. The NLOS propagation generally leads to a positive bias in the estimation range and causes a serious error in the mobile location estimation.

Nowadays many methods have been employed to mitigate the adverse effect of NLOS propagation. References [4, 5] have summarized the methods for static position systems. However, these methods are not effective for mobility tracking systems. Recently, the Kalman filtering, unscented Kalman filtering, and particle filter techniques are applied for range measurements smoothing and NLOS error mitigation [6–8]. The EKF-based algorithms are suggested in [9] as a

promising alternative to range measurement for smoothing and mitigating NLOS error. A Kalman-based IMM smoother [10] is proposed to estimate the range between the mobile node and the corresponding anchor nodes in the mixed LOS/NLOS conditions. The method in [11] proposed a one-order hidden Markov chain to simultaneously model the transition of the LOS/NLOS condition and the mobile node position.

Although the above algorithms mitigate the influence from the NLOS errors and improve the location accuracy to a certain extent, they always need to use ToA, TDoA, and RSSI to range, so it is hard to overcome the side effect from non-line-of-sight propagation. In order to further mitigate the position tracking error incurred by non-line-of-sight propagation, [12] dealt with mobile position tracking with support vector regression and game theory. To extend the above mobile tracking scheme, this paper uses support vector regressors ensemble (SVRM) to establish the map from radio parameters to node position by supervising learning. Then the position estimation from SVRM model is smoothed by the game theory.

Because the radio parameter is not considered as the distance measure, but as the feature to train the SVRM model, the side effect of non-line-of-sight is efficiently mitigated. Moreover, because of its ability to handle the uncertain and

unmodeled noise, game theory can attain theoretically more accuracy of mobile tracking compared to Kalman-filter which can only deal with the Gaussian noise.

The paper is organized as follows. Learning the map from radio parameters to node position to obtain the support vector regressors ensemble is discussed in Section 2. In Section 3, the node tracking is implemented by game theory. The simulation results are given in Section 4, and the paper is concluded in Section 5.

2. Position Estimation with Support Vector Regressors Ensemble

2.1. Position Estimation with Support Vector Regressor. Given training set, $\text{Tr} = \{(\mathbf{m}_1, p_1), (\mathbf{m}_2, p_2), \dots, (\mathbf{m}_l, p_l)\}$, where $m \in \mathbb{R}^n$ denotes radio parameters from the anchors received by mobile node, for example, ToA or RSSI, and $p \in \mathbb{R}$ the coordinates of this mobile node.

The position estimation of mobile node is to learn the regression function from radio parameters to coordinates by supervising learning. After that, if we get the radio parameter taken by location-unknown node, we can calculate its coordinates by inputting its radio parameter into this function. Usually the regression is nonlinear, so to estimate the nonlinear function between the radio feature and node coordinates, we map the data \mathbf{m} into the higher dimensional feature space \mathcal{F} where regression is linear between radio feature and node coordinates. Then in feature space linear regression can be formulated as follows:

$$p = f(\mathbf{m}) = \phi(\mathbf{m})^T \mathbf{w} + b, \quad \text{where } \phi : \mathbb{R}^n \rightarrow \mathcal{F}, \mathbf{w} \in \mathcal{F}, \quad (1)$$

where ϕ is a defined nonlinear mapping, b is a bias term, and \mathbf{w} is a coefficient vector. For support vector regression, the objective is to minimize the structure risk by estimating the weight vector \mathbf{w} and the objective function can be written as follows:

$$\min \sum_{i=1}^n g(f(\mathbf{m}_i) - p_i) + \frac{1}{2C} \|\mathbf{w}\|^2, \quad (2)$$

where $g(\cdot)$ represents ε -insensitive cost function with the following form:

$$g(f(\mathbf{m}) - p) = \begin{cases} |f(\mathbf{m}) - p| - \varepsilon, & \text{if } |f(\mathbf{m}) - p| \geq \varepsilon, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

By Lagrange multiplier technique, the objective function (2) can be expressed, after using the slack variables, as follows:

$$\begin{aligned} \max_{\alpha, \alpha^*} & \sum_{i=1}^n [\alpha_i^* (p_i - \varepsilon) - \alpha_i^* (p_i + \varepsilon)] \\ & - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i^* - \alpha_i) (\alpha_j^* - \alpha_j) \text{Ker}(\mathbf{m}_i, \mathbf{m}_j) \end{aligned} \quad (4)$$

subject to constraints

$$\sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i, \quad \alpha_i^* \leq C, \quad i = 1, 2, \dots, n, \quad (5)$$

$$\mathbf{w} = \sum_{i=1}^n (\alpha_i - \alpha_i^*) \phi(\mathbf{m}_i), \quad (6)$$

where Ker represents the kernel function with $\text{Ker}(\mathbf{m}_i, \mathbf{m}_j) = \phi(\mathbf{m}_i)^T \phi(\mathbf{m}_j)$ and α_i, α_i^* are Lagrange multipliers.

Using (6), (1) can be expressed as

$$p = f(\mathbf{m}) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) \text{Ker}(\mathbf{m}_i, \mathbf{m}) + b. \quad (7)$$

The bias b can be calculated using the point \mathbf{m}_l on the ε -margin since the prediction error for those points is known to be $\delta_l = \varepsilon \text{ sign}(\alpha_l - \alpha_l^*)$.

After solving the optimization problem (4) and calculating the b as described above, the output p that denotes the coordinates of the mobile node, corresponding to the new measurement \mathbf{m} , can be calculated by (7).

2.2. Constructing Support Vector Regressors Ensemble. To further improve the position estimation, support vector regressors ensemble is constructed by bagging. Bagging [13] is a meta-algorithm to improve classification and regression models in terms of stability and classification accuracy.

The ensemble is made of regressors built on a bootstrap sample of the training set [14]. A bootstrap sample is generated by uniformly sampling L' instances from the training set with L samples with replacement ($L' \leq L$). T bootstrap samples S_t ($t = 1, \dots, T$) are generated and the base support vector regressor is trained and built from each bootstrap. A final regressor is built whose output is weighted average of the output of the base regressors. The algorithm of bagging used in this paper is shown in Algorithm 1. The corresponding resampling subroutine is shown in Algorithm 2.

Algorithm 1 (the bagging algorithm).

Input

- (i) A training set $\text{Tr} = \{(\mathbf{m}_1, p_1), (\mathbf{m}_2, p_2), \dots, (\mathbf{m}_l, p_l)\}$, where $m \in \mathbb{R}^n$ denotes radio parameters from the anchors received by mobile node and $p \in \mathbb{R}$ the coordinates of this mobile node;
- (ii) one-against-one support vector regressor;
- (iii) integer T specifying the number of iteration; integer L' specifying the number of bootstrap samples.

Trainin Phase

For $t = 1, \dots, T$,

- (i) take a bootstrap sample S_t with sample number L' from the training set Tr using the *resampling subroutine*;

- (ii) train support vector regressor with S_t and receive the regressor f_t ;
- (iii) add f_t to the ensemble, E .

Output. For a testing set z , $p_f(z) = W_i^t \sum_{t=1}^L f_t(z)$.

Algorithm 2 (resampling subroutine).

Input. Weight vector W_i^t and training set $\text{Tr} = \{(\mathbf{m}_1, p_1), (\mathbf{m}_2, p_2), \dots, (\mathbf{m}_l, p_l)\}$.

Resampling Process

- (1) Set the data index set $I_i^t = \phi$.
- (2) Normalize $W_i^t = W_i^t / \sum_{i=1}^L W_i^t$ and compute the cumulative sum vector of W_i^t , \mathbf{C}_i .
- (3) Generate the uniformly distributed random R_i .

For $i = 1, \dots, L$, find maximum value max in \mathbf{C}_i which is less than R_i and its index in \mathbf{C}_i is j .

If max is empty, $I_i^t = 1$; else $I_i^t = j + 1$.

Output. $\text{Tr}_t = \text{Tr}_i \mid i = I_i^t$.

For a given bootstrap sample, an instance in the training set has probability $1 - (1 - 1/L')^{L'}$ of being selected at least once in the L' times instances that are randomly selected from the training set. This perturbation causes different regressors to be built, which have different certain diversities.

3. Mobile Tracking Based on Game Theory

To implement mobile node tracking, we need to smooth the node position estimate in Section 2, that is, to filter the position estimation noise. Because noise characteristics are unknown and uncertain, it is unreasonable to utilize the Kalman filter to smooth the position estimate. So we utilize the game theory to address this problem. In this scheme, we assume that the noise is the adversary of the estimator.

Assume that the mobile node measures the radio parameters at each interval ΔT that are put into support vector regression model to attain the mobile node position. Let the mobile node position estimate be $\mathbf{y}'(t) = [y'_1(t), y'_2(t)]^T$ at time t and let state vector be $\mathbf{x}'(t) = [x'_1(t), x'_2(t), x'_3(t), x'_4(t)]^T$, where $x'_1(t), x'_2(t)$ denote the x, y coordinates, respectively, of mobile location, whereas $x'_3(t), x'_4(t)$ denote the x, y coordinates, respectively, of velocity vector at time t . Then at $k\Delta T$, the state and position estimate equations can be formulated as follows:

$$\begin{aligned} \mathbf{x}'_{k+1} &= \mathbf{A}' \mathbf{x}'_k + \mathbf{O} \mathbf{u}_k + \mathbf{B} w_k + \delta_k, \\ \mathbf{y}'_k &= \mathbf{H}' \mathbf{x}'_k + \nu_k, \end{aligned} \quad (8)$$

where \mathbf{u}_k denotes the two-dimensional acceleration component, w_k is white noise sequence, and δ_k is the noise introduced by an adversary:

$$\begin{aligned} \mathbf{A}' &= \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{O} &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \Delta T & 0 \\ 0 & \Delta T \end{bmatrix}, \\ \mathbf{H}' &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}^T. \end{aligned} \quad (9)$$

Let $\mathbf{x}_k = [\mathbf{x}'_k, \mathbf{u}_k]^T$, $\mathbf{y}_k = \mathbf{y}'_k$, then the equation system of node state and position estimate can be rewritten as follows:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A} \mathbf{x}_k + \mathbf{B} w_k + \delta_k \\ \mathbf{y}_k &= \mathbf{H} \mathbf{x}_k + \nu_k, \end{aligned} \quad (10)$$

where $\mathbf{A} = [\mathbf{A}' \ \mathbf{O}]$, $\mathbf{H} = [\mathbf{H}' \ \mathbf{0}_{2 \times 2}]$, and ν_k, w_k are mutually uncorrelated unity-variance white noise sequence.

Then the mobile tracking problem is to find an estimate $\hat{\mathbf{x}}_{k+1}$ of \mathbf{x}_{k+1} given the $(\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_k)$. We assume that the estimate is unbiased and has the following structure:

$$\begin{aligned} \hat{\mathbf{x}}_0 &= 0, \\ \hat{\mathbf{x}}_{k+1} &= \mathbf{A} \hat{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k). \end{aligned} \quad (11)$$

Suppose δ_k as the noise introduced by an adversary that has the goal of maximizing the estimation error. Assume $\delta_k = \mathbf{L}_k (\mathbf{G}_k (\mathbf{x}_k - \hat{\mathbf{x}}_k) + n_k)$, where \mathbf{L}_k denotes the gain to be determined, \mathbf{G}_k is the given matrix, and n_k is the noise sequence.

The estimation error is defined as follows:

$$\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k. \quad (12)$$

It can be shown from the preceding equations that the dynamic system describing the evolution of the estimation error is given as follows:

$$\begin{aligned} \mathbf{e}_0 &= \mathbf{x}_0, \\ \mathbf{e}_{k+1} &= (\mathbf{A} - \mathbf{K}_k \mathbf{H} + \mathbf{L}_k \mathbf{G}_k) \mathbf{e}_k + \mathbf{B} w_k + \mathbf{L}_k n_k - \mathbf{K}_k \nu_k. \end{aligned} \quad (13)$$

However, this is an inappropriate term for a minimax problem because the adversary can arbitrarily increase \mathbf{e}_k by arbitrarily increasing \mathbf{L}_k . To prevent this, we decompose \mathbf{e}_k as follows:

$$\mathbf{e}_k = \mathbf{e}_{1,k} + \mathbf{e}_{2,k}, \quad (14)$$

where $\mathbf{e}_{1,k}, \mathbf{e}_{2,k}$ evolve as follows:

$$\begin{aligned} \mathbf{e}_{1,0} &= \mathbf{x}_0, \\ \mathbf{e}_{1,k+1} &= (\mathbf{A} - \mathbf{K}_k \mathbf{H} + \mathbf{L}_k \mathbf{G}_k) \mathbf{e}_{1,k} + \mathbf{B} w_k - \mathbf{K}_k \nu_k, \\ \mathbf{e}_{2,0} &= \mathbf{x}_0, \\ \mathbf{e}_{2,k+1} &= (\mathbf{A} - \mathbf{K}_k \mathbf{H} + \mathbf{L}_k \mathbf{G}_k) \mathbf{e}_{2,k} + \mathbf{L}_k n_k. \end{aligned} \quad (15)$$

From the above analysis, we can see that the estimator minimizes the objective function by searching \mathbf{K}_k , but the noise (adversary) maximizes the objective function by finding \mathbf{L}_k . Thus the estimator and noise constitute two side of a game.

For this game, we define the objective functions as

$$J(\mathbf{K}_k, \mathbf{L}_k) = \text{trace} \sum_{k=0}^N \eta_k E(\mathbf{e}_{1,k} \mathbf{e}_{1,k}^T - \mathbf{e}_{2,k} \mathbf{e}_{2,k}^T), \quad (16)$$

where η_k is any positive definite weighting matrix.

According the difference game theory, the optimal solution to the objective function (16) is the saddle point $(\mathbf{K}_k^*, \mathbf{L}_k^*)$ of difference game and satisfies the following equation:

$$J(\mathbf{K}_k^*, \mathbf{L}_k) \leq J(\mathbf{K}_k^*, \mathbf{L}_k^*) \leq J(\mathbf{K}_k, \mathbf{L}_k^*), \quad \forall (\mathbf{K}_k, \mathbf{L}_k). \quad (17)$$

For brevity, we rewrite the objective function and define the matrix \mathbf{F}_k as follows:

$$\mathbf{F}_k = \mathbf{A} - \mathbf{K}_k \mathbf{H} + \mathbf{L}_k \mathbf{G}_k. \quad (18)$$

Define the following matrix difference equation:

$$\begin{aligned} \mathbf{Q}_0 &= E(\mathbf{x}_0 \mathbf{x}_0^T), \\ \mathbf{Q}_{k+1} &= \mathbf{F}_k \mathbf{Q}_k \mathbf{F}_k^T + \mathbf{B} \mathbf{B}^T + \mathbf{K}_k \mathbf{K}_k^T - \mathbf{L}_k \mathbf{L}_k^T. \end{aligned} \quad (19)$$

Then the objective function is reformulated as follows:

$$J(\mathbf{K}_k, \mathbf{L}_k) = \text{trace} \sum_{k=0}^N \mathbf{w}_k \mathbf{Q}_k. \quad (20)$$

Now define $\widetilde{\mathbf{Q}}_k$, Σ_k as the nonsingular solution as the following set of equations:

$$\begin{aligned} \widetilde{\mathbf{Q}}_0 &= E(\mathbf{x}_0 \mathbf{x}_0^T), \\ \widetilde{\mathbf{Q}}_k (\mathbf{I} - \mathbf{H}^T \mathbf{H} \Sigma_k) &= (\mathbf{I} - \widetilde{\mathbf{Q}}_k \mathbf{G}_k^T \mathbf{G}_k) \Sigma_k, \\ \widetilde{\mathbf{Q}}_{k+1} &= \mathbf{A} \Sigma_k \mathbf{A}^T + \mathbf{B} \mathbf{B}^T. \end{aligned} \quad (21)$$

And if solutions to these equations do exist, then Σ_k can be computed as follows:

$$\Sigma_k = (\widetilde{\mathbf{Q}}_k \mathbf{H}^T \mathbf{H} - \widetilde{\mathbf{Q}}_k \mathbf{G}_k^T \mathbf{G}_k + \mathbf{I})^{-1} \widetilde{\mathbf{Q}}_k. \quad (22)$$

Theorem 3 (see [15]). *If $(\mathbf{H} \widetilde{\mathbf{Q}}_k \mathbf{H}^T + \mathbf{I}) \geq 0$ and $(\mathbf{I} - \mathbf{G}_k \widetilde{\mathbf{Q}}_k \mathbf{G}_k^T) \geq 0$, then $\mathbf{L}_k^* = \mathbf{A} \Sigma_k \mathbf{G}_k^T$, $\mathbf{K}_k^* = \mathbf{A} \Sigma_k \mathbf{H}^T$ satisfy the saddle point equilibrium (17).*

In brief, the implementation of mobile tracking is shown as follows:

(1) equation system

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A} \mathbf{x}_k + \mathbf{B} w_k + \delta_k, \\ \mathbf{y}_k &= \mathbf{H} \mathbf{x}_k + \nu_k, \end{aligned} \quad (23)$$

(2) initialization

$$\begin{aligned} \widehat{\mathbf{x}}_0 &= 0, \\ \widetilde{\mathbf{Q}}_0 &= E(\mathbf{x}_0 \mathbf{x}_0^T), \end{aligned} \quad (24)$$

(3) at each step $k = 0, 1, \dots$, do the following.

- (a) Choose the parameter \mathbf{G}_k .
- (b) Calculate the following equations:

$$\begin{aligned} \Sigma_k &= (\widetilde{\mathbf{Q}}_k \mathbf{H}^T \mathbf{H} - \widetilde{\mathbf{Q}}_k \mathbf{G}_k^T \mathbf{G}_k + \mathbf{I})^{-1} \widetilde{\mathbf{Q}}_k, \\ \widetilde{\mathbf{Q}}_{k+1} &= \mathbf{A} \Sigma_k \mathbf{A}^T + \mathbf{B} \mathbf{B}^T, \\ \mathbf{K}_k &= \mathbf{A} \Sigma_k \mathbf{H}^T, \\ \widehat{\mathbf{x}}_{k+1} &= \mathbf{A} \widehat{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{H} \widehat{\mathbf{x}}_k). \end{aligned} \quad (25)$$

- (c) If $(\mathbf{I} - \mathbf{G}_k \widetilde{\mathbf{Q}}_k \mathbf{G}_k^T) \geq 0$, then the algorithm terminates; otherwise, decrease the parameter \mathbf{G}_k and go to step (a).

4. Simulation Results

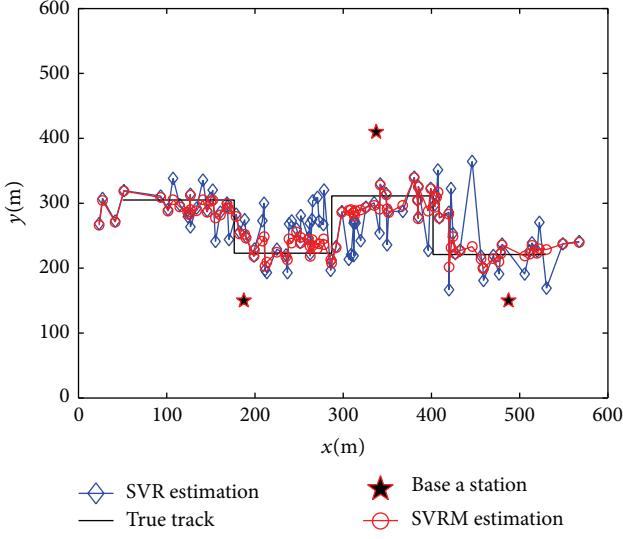
In the simulations, support vector regression is implemented by using MATLAB Support Vector Machine Toolbox. A $600 \text{ m} \times 600 \text{ m}$ relevant area with three base stations at locations (187.5, 150), (487.5, 150), and (337.5, 409.8) is considered. The mobile node is moving from location (50, 305) to (575, 230) on the solid track shown in Figure 1 with a constant speed of 10 m/s. TOA measurements are taken at a rate of 5 samples/s.

In experiment, we use elliptical scatter model [16] to produce the ToA data. For the elliptical scatter model, elliptical scatters are uniformly distributed inside the ellipse with foci at the base station and mobile. The ToA probability density function is given by the following function:

$$p(x) = \begin{cases} \frac{c(2c^2x^2 - D^2)}{4a_m b_m \sqrt{c^2x^2 - D^2}}, & \frac{D}{c} \leq x \leq t_m, \\ 0, & \text{else}, \end{cases} \quad (26)$$

where t_m is the maximum delay associated with scatterers within the ellipse, c is the speed of light, and D is the true distance between anchor and mobile node. That is, only multipath components that arrive within t_m seconds are considered. The parameters a_m and b_m are the semimajor axis and semiminor axis values which are given by $a_m = ct_m/2$, $b_m = (1/2)\sqrt{c^2t_m^2 - D^2}$.

This TOA distribution is used in the simulation by equating t_m to a multiple of the true TOA between the location of mobile node and the anchor; that is, $t_m = \alpha D/c$, so that closer locations have higher probability of having smaller NLOS errors, because the probability that signal from mobile node to anchor encounters the scatters decreases with α , that is, the distance between mobile node to anchor.



For the SVR and SVRM model, the radial basis function is used as the kernel and the ε -insensitive cost function with $\varepsilon = 1$ is employed.

In the first set of simulations, the mobile is tracked in the solid line shown in Figure 1. The location estimates obtained by the SVR and SVRM are also shown in Figure 1 with $C = 200$ in (3), $\alpha = 1.5$, and $\Delta = 30$ m, where Δ is the distance between the consecutive training locations.

The smoothing results from the Kalman filter and game theory, which take the SVR and SVRM estimates as inputs, are presented in Figure 2. From these two figures, the effect of Kalman filter and game theory is seen clearly. The average error between the true locations and the estimated locations is 40.77 m for the SVR estimates, which improves to 37.32 m after Kalman filtering and further improves to 28.08 m after smoothing by game theory [12]. But the average error between the true locations and the estimated locations is 30.58 m for the SVRM estimates and further improves to 23.08 m after smoothing by game theory. This shows that the position estimate from SVRM is better than that from SVR.

In Figure 3, the average errors after smoothing by Kalman and game theory are plotted for different values of α (from 1.5 to 2.5) with $C = 200$ and $\Delta = 30$ m. Also the estimates from the conventional least squares algorithm are plotted. As expected, the average error increases with α . However, the average error after game theory increases less fast with α than those of other methods. From Figure 3, it is also shown that smoothing SVRM output by game theory can attain more lower position tracking error than smoothing SVR output by game theory.

5. Conclusion

In this paper, we utilize SVRM to learning the map from radio parameter to node position and game theory to smooth the SVRM output. Because the radio parameter is not considered as the distance measure, but as the feature to train the

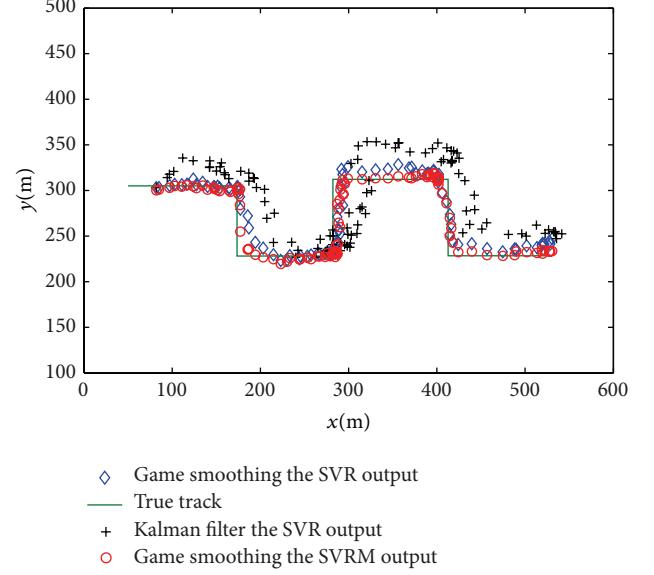


FIGURE 2: Position estimate after smoothing by Kalman filter and game.

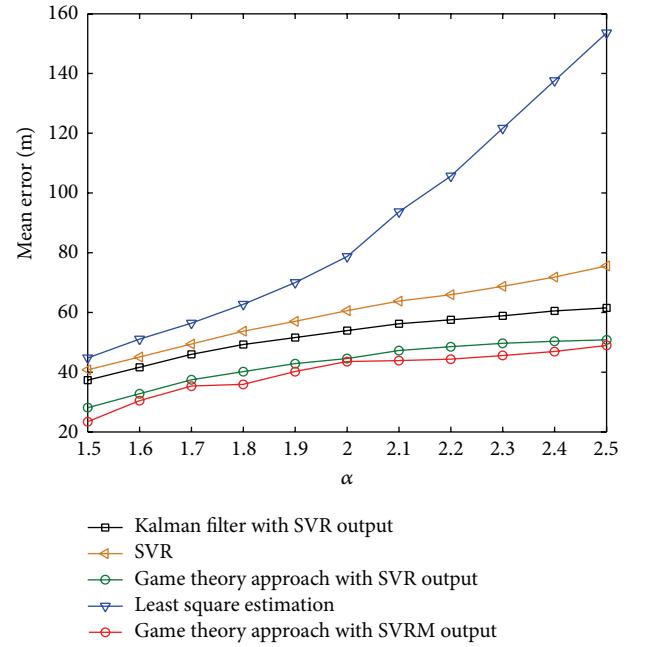


FIGURE 3: Average tracking error versus α .

SVRM model, the side effect of non-line-of-sight is mitigated. And by modeling the noise as the adversary of position estimator, because of its ability to handle the uncertain and unmodeled noise, game theory can attain more accuracy of mobile tracking compared to Kalman filter. The above method can yield very accurate position estimates even in NLOS environments and attain better performance than the method in [12]. From the simulations, it can be seen that SVRM can reduce the effect of NLOS.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grants (nos. 61370096 and 61173012), the Key Project of Natural Science Foundation of Hunan Province under Grant (no. 12JJA005), and technology project of Hunan Province (no. 2013GK3023).

References

- [1] S. Pandey and P. Agrawal, “A survey on localization techniques for wireless networks,” *Journal of the Chinese Institute of Engineers*, vol. 29, no. 7, pp. 1125–1148, 2006.
- [2] I. Amundson and D. X. Koutsoukos, “A survey on localization for mobile wireless sensor networks,” in *Proceedings of the 2nd International Workshop on Mobile Entity Localization and Tracking in GPS-Less Environments (MELT ’09)*, pp. 13–32, Orlando, Fla, USA, September 2009.
- [3] S. Gezici, “A survey on wireless position estimation,” *Wireless Personal Communications*, vol. 44, no. 3, pp. 263–282, 2008.
- [4] L. Cong and W. Zhuang, “Nonline-of-sight error mitigation in mobile location,” *IEEE Transactions on Wireless Communications*, vol. 4, no. 2, pp. 560–573, 2005.
- [5] I. Guvenc and C.-C. Chong, “A survey on TOA based wireless localization and NLOS mitigation techniques,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, pp. 107–124, 2009.
- [6] W. Ke and L. Wu, “Mobile location with NLOS identification and mitigation based on modified Kalman filtering,” *Sensors*, vol. 11, no. 2, pp. 1641–1656, 2011.
- [7] J. M. Huerta, J. Vidal, A. Giremus, and J.-Y. Tourneret, “Joint particle filter and UKF position tracking in severe non-line-of-sight situations,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 874–888, 2009.
- [8] M. Boccadoro, G. De Angelis, and P. Valigi, “TDOA positioning in NLOS scenarios by particle filtering,” *Wireless Networks*, vol. 18, no. 5, pp. 579–589, 2012.
- [9] L. Chen and L. Wu, “Mobile positioning in mixed LOS/NLOS conditions using modified EKF banks and data fusion method,” *IEICE Transactions on Communications*, vol. 92, no. 4, pp. 1318–1325, 2009.
- [10] W. Li and Y. Jia, “Location of mobile station with maneuvers using an IMM-based cubature Kalman filter,” *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4338–4348, 2012.
- [11] C. Morelli, M. Nicoli, V. Rampa, and U. Spagnolini, “Hidden Markov models for radio localization in mixed LOS/NLOS conditions,” *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1525–1542, 2007.
- [12] Z. Fanzi, L. Zhenhua, and L. Renfa, “An approach to mobile position tracking based on support vector regression and game theory,” *Journal of Computer Research and Development*, vol. 47, no. 10, pp. 1709–1713, 2010.
- [13] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [14] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, John Wiley & Sons, Hoboken, NJ, USA, 2004.
- [15] D. Simon, “A game theory approach to constrained minimax state estimation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 2, pp. 405–412, 2006.
- [16] R. B. Ertel and J. H. Reed, “Angle and time of arrival statistics for circular and elliptical scattering models,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 11, pp. 1829–1840, 1999.

Research Article

Continuous Probabilistic Skyline Queries for Uncertain Moving Objects in Road Network

Shanliang Pan, Yihong Dong, Jinfeng Cao, and Ken Chen

College of Information Science and Engineering, Ningbo University, Ningbo 315211, China

Correspondence should be addressed to Yihong Dong; dongyihong@nbu.edu.cn

Received 24 November 2013; Accepted 5 January 2014; Published 4 March 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Shanliang Pan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In moving environment, the positions of moving objects cannot be located accurately. Apart from the measuring instrument errors, movement of the objects is the main factor contributing to this uncertainty. This uncertainty makes dominant relationship of data instable, which will affect skyline operator. In this paper, we mainly study the continuous probabilistic skyline query for uncertain moving objects in road network. The query point is deemed to be stationary while moving objects are treated as targets with uncertainty described by a probability density function. After defining the notion of dominant probability and probabilistic skyline, we put forward a novel algorithm to deal with continuous probabilistic skyline query on road network. Firstly, we compute the dominant probability and skyline probability to get initial permanent p -skyline set. Then we define events to predict the time when dominant relationship between moving objects may change. Furthermore, we track and calculate events to update the probabilistic skyline in an incremental way. Two pruning strategies are proposed to cancel invalid events and objects in a bid to diminish search space. Finally, an extensive experimental evaluation on real datasets shows that probabilistic skyline sets in road network can be updated by the proposed algorithm. It demonstrates both efficiency and effectiveness.

1. Introduction

Skyline query aims to find a subset where all objects are not dominated by any other object in the dataset, helping users in multicriteria decision making, data mining and visualizing for database, and so forth.

In mobile database, a moving object reports its position and velocity to database service through wireless communication interface. Because of time delay and other technical limitations, the position obtained usually deviates from actual one. The deviation causes what is called uncertainty, which leads to instability of dominant relationship between objects. In real world, objects are restricted in road network, such as track, road, or highway. Position uncertainty of moving objects including its environment should be considered in skyline operation.

Until recently, a lot of work on skyline query had been focused on a static dataset, where the distances from query point to target objects are invariant. With rapid development of GPS technology and mobile devices, the location-based service (LBS) [1], which is pervasive in daily life, is becoming

one of the most important applications in spatiotemporal database. However, in moving environment, moving objects whose position is detected by GPS system alternately are always in motion. Delay or error of the sensor in GPS, especially the self-motion of moving objects, makes the location of moving objects uncertain. Despite the existing uncertainty, the position is often treated as being accurate to be queried in many studies [2, 3]. Study on continuous skyline query on moving objects was first proposed by Huang et al. [4]. In his work, position of moving objects is considered as the data with certainty.

In fact, because of the sensors' error and the objects' movement, the position of objects in moving environment is uncertain and vague. Recently, a few researchers have been aware of this imprecision and made some contributions [5–7]. In our approach, the location of query object is exactly known as a stationary object, but the location of moving target objects is characterized by a certain distribution instead of a precise point. For example, a weapon-related crime takes place somewhere, which is located in point O , as Figure 1 shows. Police cars should be dispatched to intervene. The

adequate police strength including the number of policemen and equipment in the dispatched car is needed. Also a car as near to the crime scene as possible is another factor for consideration in order to arrive in time. In Figure 1, table shows the parameters of each police car, including position of the cars, number of policeman, equipment level, and network distance to the query point O , where the value of equipment level is used to quantify the level of equipment, with greater value corresponding to higher level. If the police cars are treated as certain points, cars A and E are skyline objects. However, the cars are moving quickly, so their positions are uncertain to some extent. For example, suppose that the true position of car A is at point A' , where the distance from A' to O is 130 instead of 125, and car B 's actual position is at point B' whose distance to O is 125. In this case, car B can dominate car A , so car A is not skyline object. Because of road network limitation, the uncertain area is represented as a line centered on its acquired location, as Figure 1 shows. In order to simplify this scene, probability density function $f(t)$ assumes uniform distribution and the police car B is possible to be skyline object.

In this paper, we address the problem of continuous probabilistic skyline query for uncertain moving objects in road network. In our study case, the query object stays still, while the target objects are moving with uncertainty which is characterized by both a closed uncertainty region and a probability density function (pdf). The main contributions are as follows.

- (1) Probabilistic skyline on uncertain moving object in road network is introduced as a new important issue in decision support or navigation systems.
- (2) By analyzing the dominant relationship between moving objects in uncertain position environment, the dominant probability and skyline probability are defined in case the query is stationary with targets moving.
- (3) Based on trigger events which are presented to track the change of dominant relationship, a novel algorithm Probabilistic Skyline query with Uncertainty in Road network (PSUR) is proposed to deal with the dynamic skyline query with uncertainty. A series of pruning strategies are introduced to optimize and fasten this incremental algorithm.
- (4) A great deal of experiments on two truth datasets are conducted to analyze the uncertain region, numbers of static dimensions, velocity of moving objects, and dataset size that affect the algorithm. Contrast experiments from literature [8] are made to validate the proposed PSUR. The experimental results show that PSUR is effective and efficient.

The rest of this paper is organized as follows. In Section 2, we summarize the related work of skyline computation. In Section 3, after describing basic notion of skyline and uncertain model for moving objects, we define dominant probability and probabilistic skyline. Section 4 introduces trigger events to track how dominant probability relationship

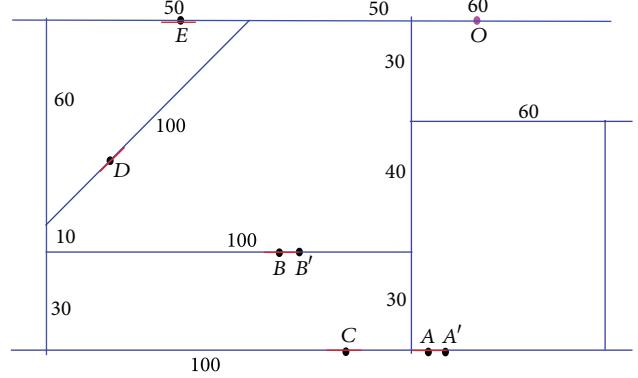


FIGURE 1: An example of skyline query for uncertain data.

varies when objects are moving with uncertainty. Two pruning strategies are adopted to reduce the search space. Section 5 describes PSUR algorithm to update p -skyline set. Section 6 gives experimental evaluation on two real datasets. Finally, the conclusion is reached in Section 7.

2. Related Work

Skyline queries are hot areas of current database research which recently have attracted more and more attention. It is introduced into relational database firstly by Borzsonyi et al. [9] with two proposed processing algorithms Block-Nested-loops algorithm (BNL) and Extended Divide & Conquer algorithm (D&C). As an improved method of BNL proposed by Dr. Chomicki et al. Sort-Filter-Skyline algorithm (SFS) [10] constructs the multidecision dominative order chains for the ordered data. NN algorithm [11], based on R-tree, searches the nearest neighbor recursively. It speeds up the skyline query by reducing the comparison numbers of BNL among objects. Nevertheless, it will cost more time and space in searching subspace repeatedly. Overcoming this limitation, the sorted R-tree based BBS [12] is proposed. It is one of the best skyline query methods in centralized datasets.

In recent years, skyline query has been extended to the dynamic datasets. The R-tree based I-Eager and I-Lazy algorithms were brought forth by Tao and Papadias [13] who firstly studied how to update and maintain the skyline results in dynamic datasets.. Tian et al. [14] introduced GICSC updating skyline query sets dynamically which better fits for low dimension metrics. Kontaki et al. [15] exerted efforts to maintain k -dominant skyline objects with maximum user's preference. Huang et al. [4] probed into the continuous skyline query for certain moving objects. It assumes that all points including the query point move in a predictable way. After analyzing dominating relationship between points in the space, Huang presented a continuous tracking algorithm-CSQ to maintain skyline sets dynamically.

In sensor networks and moving environment, the characters of the objects are not exactly known due to the limitations of measuring equipment and objects' movement. A lot of work has focused on uncertain data. The authors in [5] studied the execution of probabilistic range and nearest-neighbor

queries in mobile environment. The author in [16] focused on the situation in which the location of a query object is not exactly known. In research of skyline query field, Fiedler [17] firstly proposed skyline operator on uncertain data in his dissertation. Pei et al. [18] proposed a probabilistic skyline model for multiinstance data, where each object is part of the skyline with a certain probability. They presented two algorithms BUM and TDM to study skyline query on uncertain data of the possible world model (PWM). Lian and Chen [19] studied reverse-skyline query. They modeled the probabilistic reverse skyline query on uncertain data, in both monochromatic and bichromatic case, and proposed two effective pruning methods, MPRS and BPRS, to reduce the search space of query processing. Zhang et al. [20] explored how to maintain skyline sets when uncertain data are updated. An AR-tree is constructed for all uncertain data, and the maintenance is inserting, deleting, and updating on this AR-tree. The authors in [21] designed partitioning method to compute skyline probabilities for discrete data with uncertainty. The authors in [8] investigated skyline probabilities with a parametric form pdf (e.g., a Gaussian function or a Gaussian mixture model). The authors in [22] proposed a sliding window skyline model to study the execution of the probabilistic skyline query over uncertain data streams [23]. The authors studied a new problem of range-based skyline queries. Two novel algorithms I-SKY and N-SKY were presented to solve the probabilistic and continuous range-based skyline queries.

In road network, Huang and Jensen [24] assumed that the user's movement is constrained to a road network. The authors defined route nearest-neighbour skyline queries to consider the computation efficiency. Deng et al. [25] studied multisource skyline query in road networks. Three different road networks of multisource query methods were presented. The authors in [26] proposed a new method to process continuous skyline in road network based on precomputing the shortest range data of targets. The authors in [27] introduced route skyline computation in a multiattribute graph. Top routes are computed iteratively in an efficient way and pruning technique is adopted in order to reduce the search space. The authors in [28] focused on extracting the path skylines and proposed PathSL to generate an optimal skyline for moving objects.

In spite of much work focusing on uncertainty of skyline queries or route skyline queries, there is little work completed for skyline query concerning uncertainty of moving objects. The authors in [4, 22–25] ignored the uncertainty of moving objects, while the authors [8, 18, 21] dealt with discrete data with independent dominant relationship between objects. It is the first time to compute continuous probabilistic skyline query with regard to uncertainty of moving objects in road network.

3. The Probabilistic Skyline for Uncertain Moving Objects

3.1. Skyline on Certain Points. Suppose that the point set is $S = \{s_1, s_2, \dots, s_{\text{Num}}\}$, where Num is the number of points, each point in an n -dimensional numeric space

TABLE 1

Position of police car	A	B	C	D	E
Number of policemen	6	6	6	4	5
Equipment level	3	3	2	3	2
Network distance to O	125	130	140	155	90

$D = \{d_1, d_2, \dots, d_n\}$. The original concept of skyline is based on the notion of dominance. Let $X = (x_1, x_2, \dots, x_n)$, $Y = (y_1, y_2, \dots, y_n)$, $X \in S$, $Y \in S$, if $\forall i$, $x_i \leq y_i$ ($1 \leq i \leq n$), and $\exists j$, satisfy $x_j < y_j$ ($1 \leq j \leq n$), then X is said to dominate Y , denoted by $X \prec Y$. Given a set of objects S , an object X is a skyline point if there is not any other point dominating X . The skyline on S is the set of all skyline points.

3.2. Uncertain Model for Moving Objects. After Wolfson et al. [1] firstly studied the uncertainty of moving objects, a lot of work has focused on this field [5–7]. An uncertain region [5] of a moving object M_i at time t , denoted by $Ui(t)$, is a closed irregular region with a velocity \vec{v} such that the recorded location O_i can be found only inside this region. The pdf represents the probability density distribution of an object in its uncertain region. The uncertainty pdf [5] of an object M_i , denoted by $f(o_i)$, is a pdf of O_i , that has a value of 0 outside $Ui(t)$.

3.3. The Dominant Probability. What we mainly study on is as follows. There is a set of moving target objects whose centers are $O = \{o_1, o_2, \dots, o_{\text{Num}}\} \in R^n$ and a stationary object q as a query point to continuously compute updated probabilistic skyline dataset.

Dissimilar to the traditional skyline query, in moving environment, the spatial location is changing with time. All attributes are divided into dynamic attributes and static ones. Let us suppose that there are m dynamic attributes, k static attributes, and n attributes where $n = m + k$. For each object $O_i = (x_1^i, x_2^i, \dots, x_d^i)^T$, the static attributes construct a vector denoted by $O_i^S = (x_{S1}^i, x_{S2}^i, \dots, x_{Sk}^i)^T$, while the dynamic ones are made up of a vector $O_i^D = (x_{d1}^i, x_{d2}^i, \dots, x_{dm}^i)^T$, where $n = k + m$. For instance, in Figure 1, the number of police and equipment level belong to static attributes because they are invariant in our study case. However, the distance from O_i to q is varying continuously with police car O_i moving. In order to simplify the description, dynamic attributes are supposed to include only spatial position (Table 1).

Definition 1 (dominant probability). Let O_i, O_j be two moving objects with uncertainty and q be a stationary query point. Then probability that O_j dominates O_i is

$$\begin{aligned} & p(o_j \prec o_i) \\ &= \int_{Ur_{O_j}} \int_{Ur_{O_i}} f_{O_i}(\vec{x}_i) f_{O_j}(\vec{x}_j) \text{ isdom}(\vec{x}_j, \vec{x}_i) do_i do_j, \end{aligned} \quad (1)$$

where $f_{O_i}(\vec{x}_i)$ and $f_{O_j}(\vec{x}_j)$ are probability density function of object O_i and O_j , $\vec{x}_j \in Ur(O_j)$, $\vec{x}_i \in Ur(O_i)$, and function

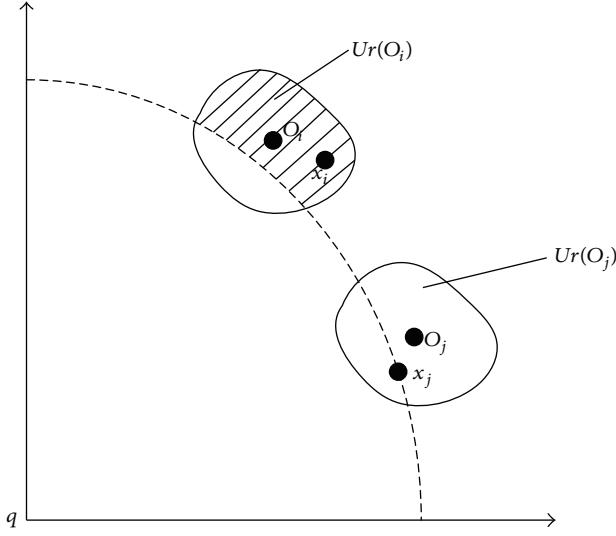


FIGURE 2: Dominant probability.

$$\text{isdom}(\vec{x}_j, \vec{x}_i) = \begin{cases} 1 & \text{dist}(\vec{x}_j, q) \leq \text{dist}(\vec{x}_i, q) \\ 0 & \text{otherwise} \end{cases}, \quad \text{dist}(\vec{x}_j, q) \text{ is Euclidean distance between } x_j \text{ and } q.$$

According to the definition of dominance in (1), we maintain that O_j can have dominance probability on O_i only if the static attributes of O_j dominate the static ones of O_i . As shown in Figure 2, for arbitrary point x_j in $Ur(O_j)$, each point x_i in $Ur(O_i)$ that satisfies $\text{dist}(\vec{x}_j, q) \leq \text{dist}(\vec{x}_i, q)$ will be dominated by x_j , as the hatched area shows.

3.4. The Skyline Probability

Definition 2 (skyline probability). The skyline probability of object O_i is the likelihood that object O_i is not dominated by any other object and is defined below:

$$\begin{aligned} \Pr(o_i) &= \Pr\left(\bigwedge_{O_j \in O} O_j \nless O_i\right) \\ &= \int_{Ur_{O_i}} f_{O_i}(\vec{x}_i) \\ &\quad \times \prod \left(1 - \int f_{O_j}(\vec{x}_j) \text{isdom}(\vec{x}_i, \vec{x}_j) d\vec{x}_j\right) d\vec{x}_i, \end{aligned} \quad (2)$$

where $f_{O_i}(\vec{x}_i)$ and $f_{O_j}(\vec{x}_j)$ are probability density function of object O_i and O_j , respectively, $\vec{x}_j \in Ur(O_j)$, $\vec{x}_i \in Ur(O_i)$.

Definition 3 (p -skyline). Let $p \in [0 \dots 1]$ be a threshold. The p -skyline is the set of objects for which the following property holds:

$$S_p = \{O_i \in O \mid \Pr(O_i) \geq p\}. \quad (3)$$

3.5. Uncertain Model in Road Network. The road network can be treated as a nondirection graph $G = \langle V, E, W \rangle$, where V is node set representing crossroad, E is edge representing roads between two crossroads, and W is length of E .

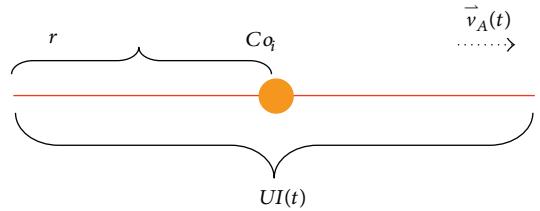


FIGURE 3: Uncertain model in road network.

In road network, the objects are limited movement, so the uncertain domain is denoted by a line segment with $f(t)$ distribution, which $f(t)$ is pdf, as Figure 3 shows.

Denote $d(v, v')$ by the shortest path between nodes v and v' . $d(v, v') = \infty$ if there exists no path from v to v' . In road network, the distance is represented by shortest path.

Definition 4 (minimum distance function). The minimum distance between uncertain object O_i to query q at time t is denoted by $d_{\min}(q, o_i) = d(q, o_i) - r$.

Definition 5 (maximum distance function). The maximum distance between uncertain object O_i to query q at time t is denoted by $d_{\max}(q, o_i) = d(q, o_i) + r$.

In continuous movement, the dominant relationship of two objects will change. For two moving objects O_i and O_j ,

$$p(o_j \prec o_i) = \begin{cases} 1 & d_{\max}(q, o_j) \leq d_{\min}(q, o_i) \\ 0 & d_{\min}(q, o_j) \geq d_{\max}(q, o_i) \\ \int_{Ur_{O_j}} \int_{Ur_{O_i}} f_{O_i}(\vec{x}_i) f_{O_j}(\vec{x}_j) \text{isdom}(\vec{x}_j, \vec{x}_i) d\vec{x}_i d\vec{x}_j & d_{\min}(q, o_j) < d_{\max}(q, o_i), \\ & d_{\max}(q, o_j) > d_{\min}(q, o_i). \end{cases} \quad (4)$$

The maximum and minimum distance functions maybe intersect with each other among moving objects. Two distance functions of object O_i and O_j are given in Figure 4. The maximum distance of O_i is less than the minimum distance of O_j prior to time t_1 , so O_j cannot dominate O_i . O_j might begin to dominate O_i from t_1 to t_2 , as the hatched area shows, so does it from t_3 to t_4 , and from t_5 to t_6 .

4. Tracking by Events

The probabilistic skyline set is updated in an incremental way. The key step is how to predict the time when the dominant relationship changes. It's hard to know the accurate time when p -skyline just change. But we can estimate period of time during which the p -skyline may change.

4.1. Trigger Events. As shown in Figure 4, from time t_1 to time t_2 , the dominant relationship between O_i and O_j might change, because the maximum distance of O_i is greater than the minimum distance of O_j . So we call the time between

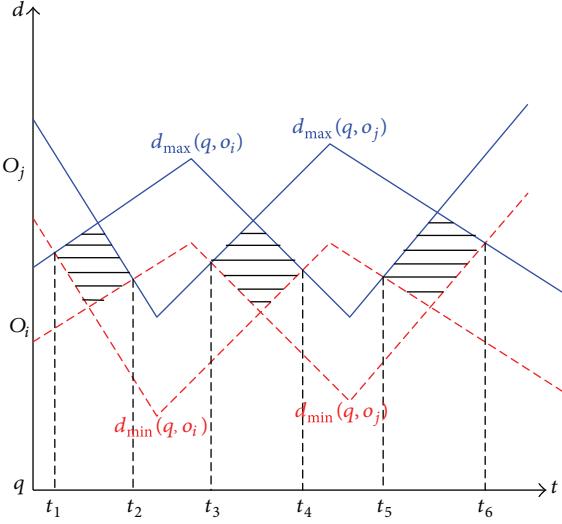


FIGURE 4: Transformation of dominant relationship.

t_1 and t_2 an event. An event on O_i can be presented as $\text{event}(o_i, o_j) = \langle o_i, o_j, s_time, e_time \rangle$, where O_i and O_j are two objects whose dominant relationship may be varied when s_time and e_time are start time and end time of the event, respectively.

4.2. The Pruning Strategy for Events. By analyzing relationship of objects we know that if static characters of any two objects O_i and O_j have dominant relationship, supposing that $O_j^S \prec O_i^S$, an event maybe occur. With the query object moving, a considerable amount of events will occur. In order to improve effectiveness of our algorithm, we propose a series of event pruning strategies.

Pruning Strategy 1. If the dominant relationship in static attributes between two moving objects O_i and O_j does not exist, the intersection of those two distance functions will cause no variation to p -skyline set. In this case, the intersection will not cause any event.

Proof. Suppose $\vec{o}_i = (x_1^i, x_2^i, \dots, x_d^i)$, $\vec{o}_j = (x_1^j, x_2^j, \dots, x_d^j)$, whose static characters are $\vec{o}_i^S = (x_{s1}^i, x_{s2}^i, \dots, x_{sk}^i)$ and $\vec{o}_j^S = (x_{s1}^j, x_{s2}^j, \dots, x_{sk}^j)$, respectively. It is known that the static characters of O_i and O_j do not possess any dominant relationship, so $\exists k_p, k_q, x_{k_p}^i > x_{k_q}^j$ and $x_{k_q}^i < x_{k_p}^j$. Even considering their dynamic attributes, the dominant relationship does not exist at any time. In conclusion, if the dominant relationship in static attributes between two objects does not exist, the intersection of their distance functions will not cause any change for their dominant probability. Events cannot take place. \square

Pruning Strategy 2. Suppose $\exists o_i, o_j, p(o_j \prec o_i) = 1$. $\text{Event}(i, j, t_{\begin{array}{l} \text{begin}, t_{\end{array}}})$ is an event interrelated to O_i and O_j . If $\exists \text{event}(O_i, O_k, t'_{\begin{array}{l} \text{begin}, t'_{\end{array}}})$ ($0 < k < \text{Num}$ and $k \neq i$)

and $t'_{\begin{array}{l} \text{begin}, t'_{\end{array}}} < t_{\begin{array}{l} \text{begin}}$, the computation for $\text{event}(O_i, O_k, t'_{\begin{array}{l} \text{begin}, t'_{\end{array}}})$ is invalid before $t_{\begin{array}{l} \text{begin}}$. However, if $t'_{\begin{array}{l} \text{end}, t'_{\end{array}}} < t_{\begin{array}{l} \text{begin}}$, $\text{event}(i, k, t'_{\begin{array}{l} \text{begin}, t'_{\end{array}}})$ is also invalid.

Proof. Because $p(o_j \prec o_i) = 1$, it is known that $p(o_i \prec o_j) = 0$ prior to time $t_{\begin{array}{l} \text{begin}}$. The update for $\text{event}(O_i, O_k, t'_{\begin{array}{l} \text{begin}, t'_{\end{array}}})$ is invalid. It is also invalid when $t'_{\begin{array}{l} \text{end}, t'_{\end{array}}} < t_{\begin{array}{l} \text{begin}}$. \square

5. Continuous Probabilistic Skyline Queries Algorithm

5.1. Initialization. The initialization framework is presented in this section. Each object has static and dynamic attributes, so we should compute the dominance relation on static attributes firstly. For two objects O_i and O_j , O_j might dominate O_i only when $o_j^S \prec o_i^S$ (Algorithm 1). In order to compute fast, static skyline set denoted by S_{static} is introduced if only the dominant relationship for static characters is considered.

In any case objects move, S_{static} always belongs to skyline. In initialization step, the moving path and distance function of each moving object need to be precomputed according to the information of road network and moving objects, apart from S_{static} .

5.2. PSUR Algorithm. Not all skyline probability of objects will change at one moment. If maximum/minimum distance function of one object cuts that of others in Cartesian coordinates, skyline probability of this object maybe vary (Algorithm 2). In other words, trigger events include all possible variations of p -skyline for each moving object.

In order to simplify the problem, we suppose that all moving objects preserve their velocity with uniform speed. If not, the precomputing cannot be processed in advance. All events should be recomputed again. The movement of moving objects to query point can be picked up with its shortest path to query point. The intersecting time for the distance function can be recomputed and put into event queue sorted by time in ascending order. For each time, skyline probabilities of moving objects under trigger events need to be updated.

5.3. Algorithm Analysis and Discussion. The cost incurred by our method consists of three components: initialization, events computing, and updating by tracking.

In initialization period, computing static dominant relationship will cost $O(d \cdot n(n - 1)/2)$, where d is static dimensional degree. The most cost is skyline probability computation. The famous Simpson integration method, which is introduced to compute skyline probability of each moving objects Num, and time interval t . Probabilistic $O((\text{Len}/h)^2)$, where Len is uncertain segment length and h is step width of Simpson, so the cost of skyline probability of each moving object is $O((\text{Len}/h)^2(n-1))$. After skyline probability computation, judgment for p -skyline objects will cost $O(n)$. Above

```

Input: one query object,  $n$  moving objects;
Output: Initialization  $p$ -skyline;
(1)  $S_{\text{static}} = \text{NULL}$ 
(2) for any point  $o_i$  in  $O$ 
(3) compute_path( $o_i$ ); //get the moving path of objects
(4) compute_distance( $o_i$ ); //computing distance function
(5)  $O_i = \text{NULL}$ 
(6) for any point  $o_j$  ( $j \neq i$ ) in  $O$ 
(7) compare( $o_j^S, o_i^S$ ); //ensure whether  $O_j$  dominate  $O_i$  in static attributes;
(8) if  $o_j^S < o_i^S$ 
(9) flag( $o_j, o_i$ ) = 1;
(10) push  $o_j$  into the  $O_i$ ;
(11) else
(12) flag( $o_j, o_i$ ) = 0;
(13) if the  $O_i = \text{NULL}$ 
(14)  $S_{\text{static}} = S_{\text{static}} \cup o_i$ ;

```

ALGORITHM 1: Initialization.

```

Input: the set of moving objects  $D$ , query point  $q$ ,  $p$  is the probabilistic threshold, the time interval  $[t_s, t_e]$ .
Output: the  $p$ -Skyline at any time instant  $t$  within  $[t_s, t_e]$ .
(1) initialization();
(2) // compute events;
(3) for any point  $o_i$  in  $O$ 
(4) for each point  $o_j$  in  $O_i$ 
(5) compute begin time  $s\_time$  and end time  $e\_time$  of each event;
(6) if  $s\_time \leq t_e \&& e\_time \geq t_s$ 
(7) push it into  $Q_e$ ; //valid event;
(8) //handle events;
(9)  $time = t_s, L_p = L_s$ 
(10) while ( $time \leq t_e$ )
(11) while ( $Q_e \neq \text{NULL}$ )
(12)  $e = Q_e \cdot \text{erase}()$ ; // pop queue's head;
(13) while ( $e \cdot t_{\text{begin}} \leq time$ )
(14)  $Q_{\text{handle}} \cdot \text{push\_back}(e)$ ; // push  $e$  into queue;
(15)  $e = Q_e \cdot \text{erase}();$ 
(16) handle( $Q_{\text{handle}}$ ); //update  $Q_e$  and  $S_p$ 
(17) time++;

```

ALGORITHM 2: PSUR algorithm.

all, time cost in initialization is $O(d \cdot n(n-1)/2 + (\text{Len}/h)^2(n-1) + n)$.

In second step, the worst cost of comparison for static dominant relationship between objects is $O(n(n-1)/2)$. The interaction time for arbitrary two moving objects' distance function will cost $O(\delta_1)$, where δ_1 is constant. Thus it will cost $O(\delta_1 \cdot n(n-1)/2)$ in this period.

Handling events generates at most $n(n-1)/2$. In the worst case, each event is valid in time period $[t_s, t_e]$, and this requires $(t_e - t_s)$ times to deal with events. In every processing, skyline probability related to trigger events will cost $O((\text{Len}/h)^2(n-1))$, while update computing for event is $O(\delta_2)$, where δ_2 is constant. The total cost in this period is $O((\text{Len}/h)^2(n-1) \cdot (t_e - t_s) \cdot n(n-1)/2 + \delta_2 \cdot (t_e - t_s) \cdot n(n-1)/2)$.

In conclusion, the total cost is added together for these three periods, which is equivalent to $O(n^2)$.

6. Experimental Evaluation

6.1. Datasets. Two real road networks are used to test the effect and efficiency of the proposed algorithm PSUR. One is the famous seashore city Oldenburg in Germany, which includes 6105 nodes and 7036 edges. The other is Cixi city of China, which contains 244 intersection nodes and 407 edges, much smaller than the first one. We assume that the uncertain area is segment along the road. Two distributions of uniform and Gaussian distribution for probability density function are adopted because these two functions are by far the most important and commonly used in statistics. Moving objects are generated randomly. Baseline and priority algorithms proposed in [8] are used to compare with our method PSUR to verify efficiency and effect. The main parameters are segment length Len, static dimension number d , numbers of

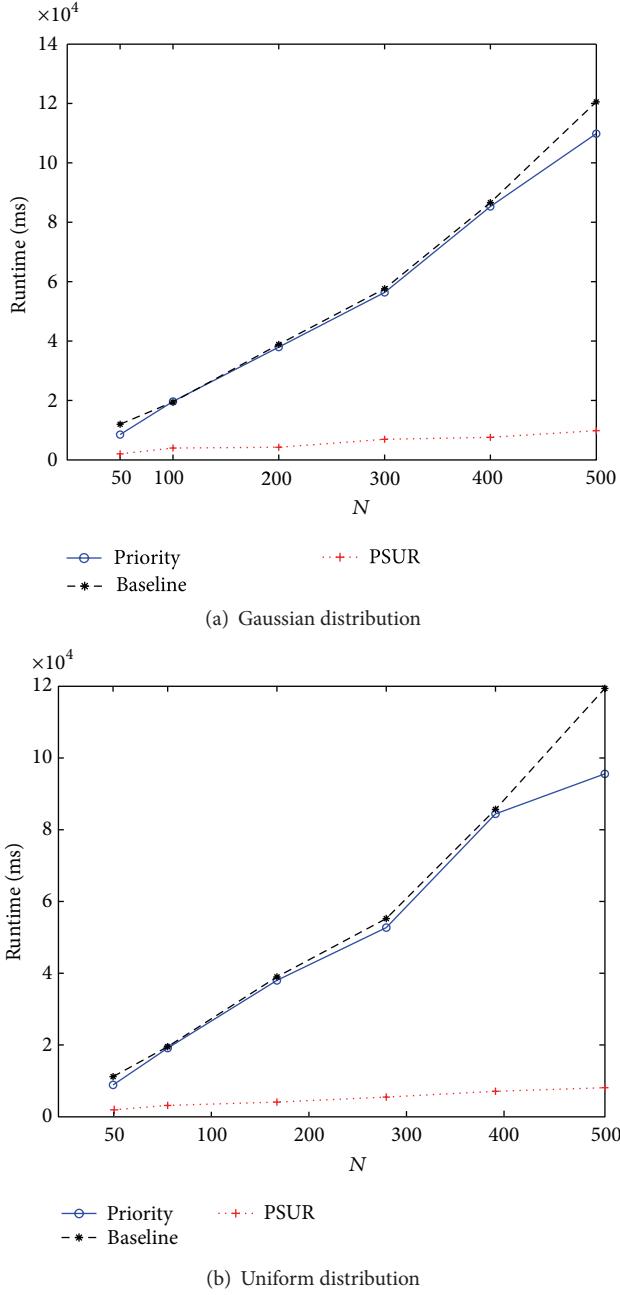


FIGURE 5: Numbers of moving objects effects of the performance in Cixi city ($d = 5$, $\text{Len} = 10$ and $t = 20$).

moving objects Num, and time interval t , and probabilistic threshold p is 0.5 in all experiments.

We conducted our experiments on desktop PC running on Windows XP professional. The PC has Intel Core 2Duo 2.93 GHz and 3 GB RAM memory. All experiments were coded in Visual C++ 2008.

6.2. Numbers of Moving Objects. In this experiment, suppose $d = 5$, $\text{Len} = 10$, and $t = 20$. Figure 5 shows performance of PSUR versus baseline and priority [8] when numbers of moving objects vary on Cixi's road network, both in

TABLE 2: Event size in Cixi road network ($d = 5$, $\text{Len} = 10$, $t = 20$).

Numbers of moving objects	Event size before pruning	Event size after pruning
50	669	35
100	2721	95
200	10389	205
300	35009	400
400	62723	514
500	96890	736

TABLE 3: Event size in Oldenburg road network ($d = 5$, $\text{Len} = 10$, $t = 20$).

Numbers of moving objects	Event size before pruning	Event size after pruning
50	245	5
100	455	15
200	1557	41
300	3945	131
400	6464	236
500	9951	367

uniform distribution and Gaussian distribution. Likewise, Figure 10 demonstrates the performance on road network in Oldenburg. The figures show that runtime increases with object's numbers whichever the model is. No matter what model is used, uniform or Gaussian, the cost in the same dataset is similar. Among these three methods, the response time of our algorithm is only one-tenth of the other two methods. It is because baseline and priority need recompute the probabilistic skyline set at each time while PSUR is an incremental method.

If there is the same number of moving objects in two real networks, the density of moving objects in Cixi is bigger than that in Oldenburg, because Cixi is smaller than Oldenburg. Therefore the chances for interaction of moving objects in Cixi are more those that in Oldenburg. Trigger events generated in Cixi are evidently more than those in Oldenburg, as Tables 2 and 3 show. The density of moving objects leads to this difference, because it is more sparsely distributed in Oldenburg than in Cixi for the same number of objects. As a result, by observing the performance between two city networks under the same data model, such as Figures 5(a) and 6(a), it is noted that the runtime in the smaller city Cixi costs a little more than that in Oldenburg.

The pruning strategy has done effective work on event size, as Tables 2 and 3 show. Event size will grow with the number of moving objects, because the number of the intersection of distance function rises.

6.3. The Effect of Uncertain Segment Length. In this section, parameters are as follows: $\text{Num} = 50$, $d = 5$ and $t = 20$. Figures 7 and 8 show performance of three algorithms when segment length of uncertainty varies on Cixi road network

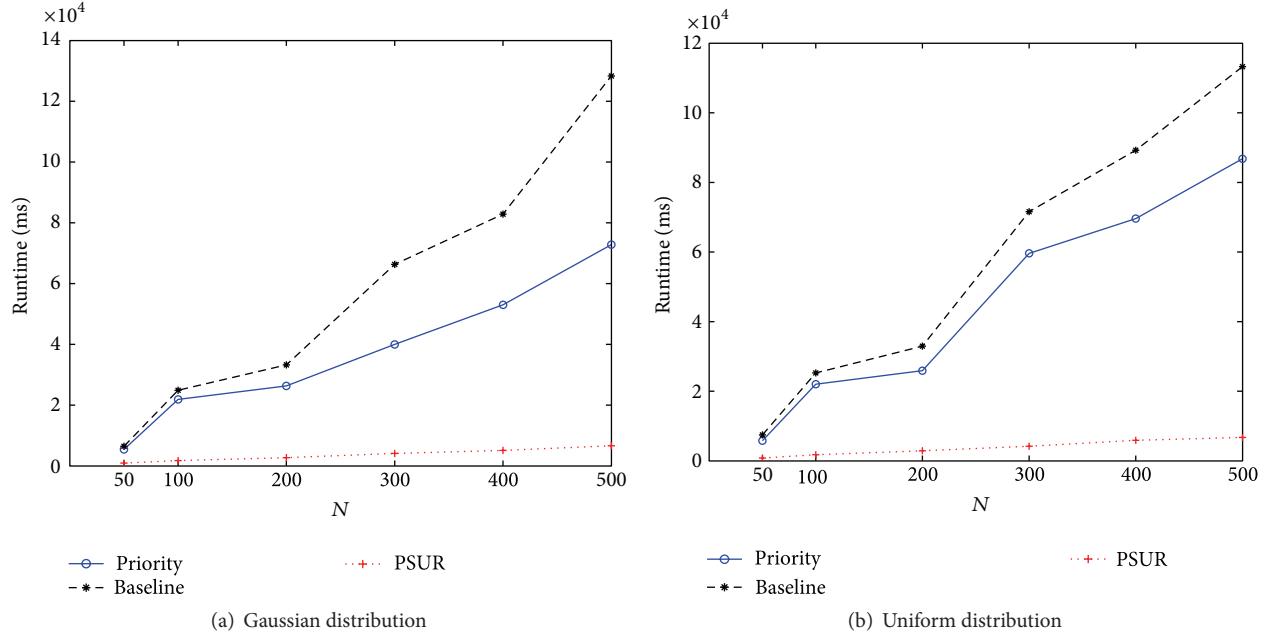


FIGURE 6: Numbers of moving objects effect of the performance in Oldenburg city ($d = 5$, Len = 10 and $t = 20$).

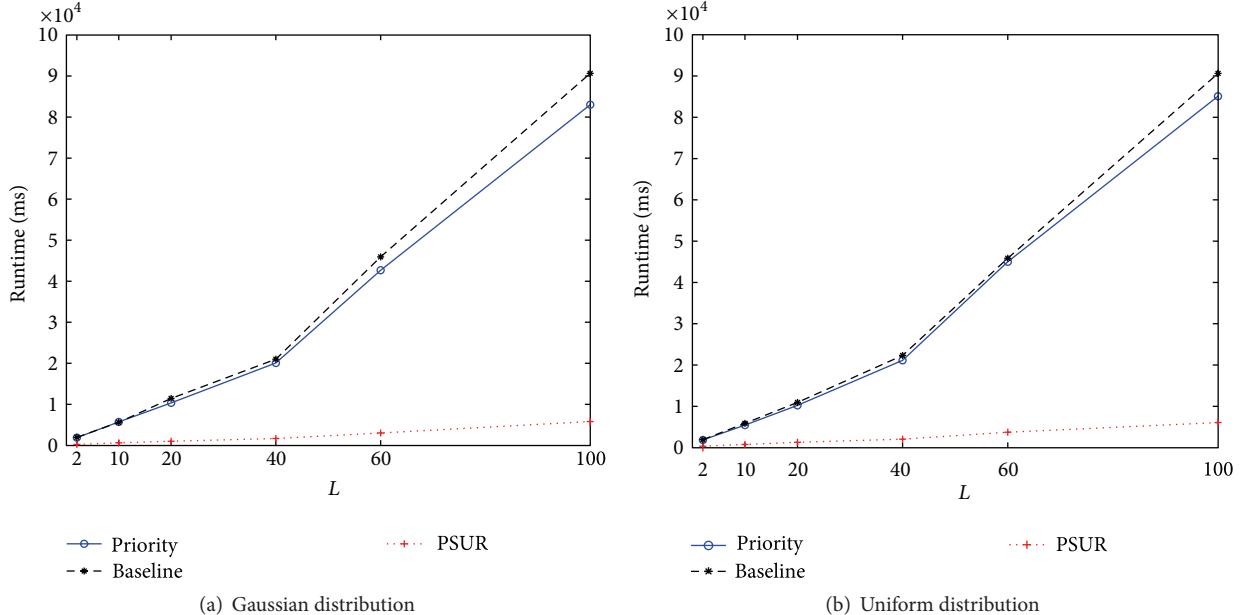


FIGURE 7: Segment length effects of the performance in Cixi city (Num = 50, $d = 5$, $t = 20$).

and Oldenburg city, respectively, both in uniform distribution and Gaussian distribution.

The magnitude of the uncertain length may also affect the performance of algorithm PSUR. It is seen that the longer the segment length is, the more runtime is required. If the length becomes much longer, for any two objects O_i and O_j , the probability that $\Pr(O_j < O_i) \neq 0$, $\Pr(O_j < O_i) \neq 1$ happen becomes greater. According to the definition of event, the size of event queue will increase, so will the times to track and handle events. The calculation of dominance probability and

skyline probability grows with this increasing probability, so that the cost will go up.

It is shown that the segment length of uncertainty affects event's size, as shown in Tables 4 and 5. With increase of segment length, the number of events becomes great.

6.4. The Effect of Static Attribute Dimensionality. We will discuss the effect that multidimensional attributes impact on our algorithm over real road network. The result is shown in Figures 9 and 10. It is known that the higher dimension

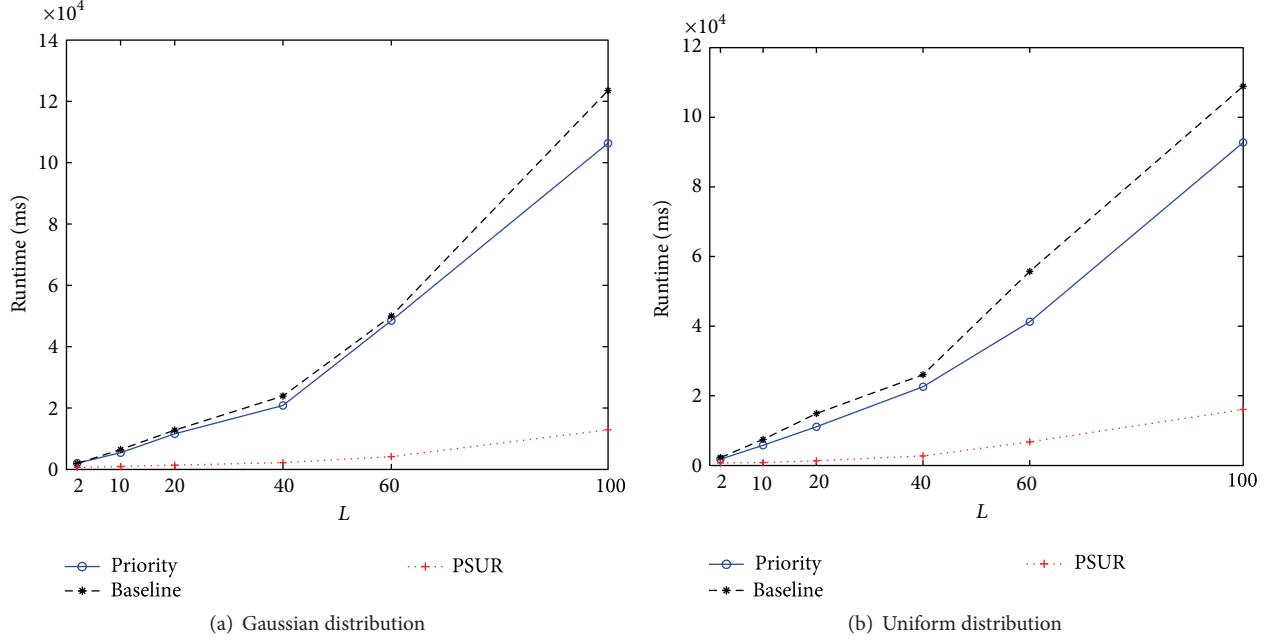


FIGURE 8: Segment length effects of the performance in Oldenburg city (Num = 50, d = 5, t = 20).

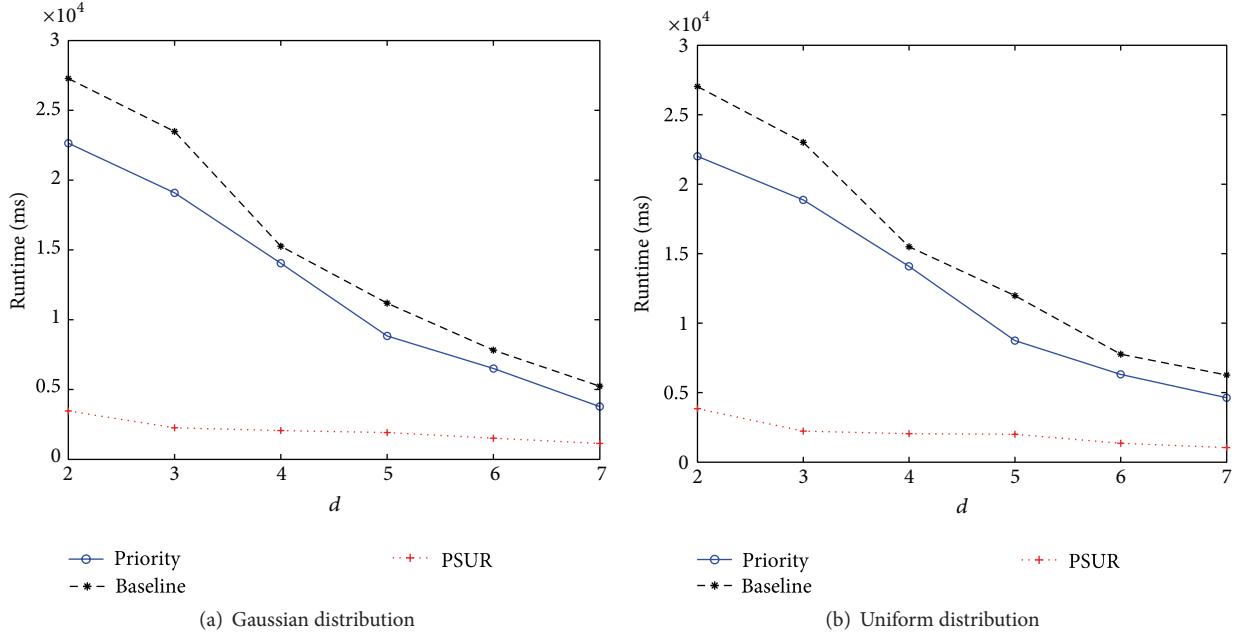


FIGURE 9: Static attribute dimensionality effects of the performance in Cixi city (Num = 50, Len = 20, $t = 20$).

results in less run-time. This is because the computation of dominance relation on static attributes is carried out only once at the beginning. The higher the static dimensions are, the less objects that dominant o_i are. The less the time cost on computation of initializing adjacency list and event queue is, the less the total runtime is.

Tables 6 and 7 show the event size changes with static dimensionality of PSUR. Unlike preceding two experiments, in this case, when static dimensionality d increases, the event

size reduces instead of increasing. This is due to the fact that, when d increases, the number of moving objects which have dominant relationship in static dimensionality decreases. The distance function will interact less, so will valid events.

6.5. The Effect of Time Span. Baseline and priority are somewhat static algorithms. They need to recompute skyline probability for each moving object, so runtime is proportional to the time span. However, as a dynamic method, PSUR

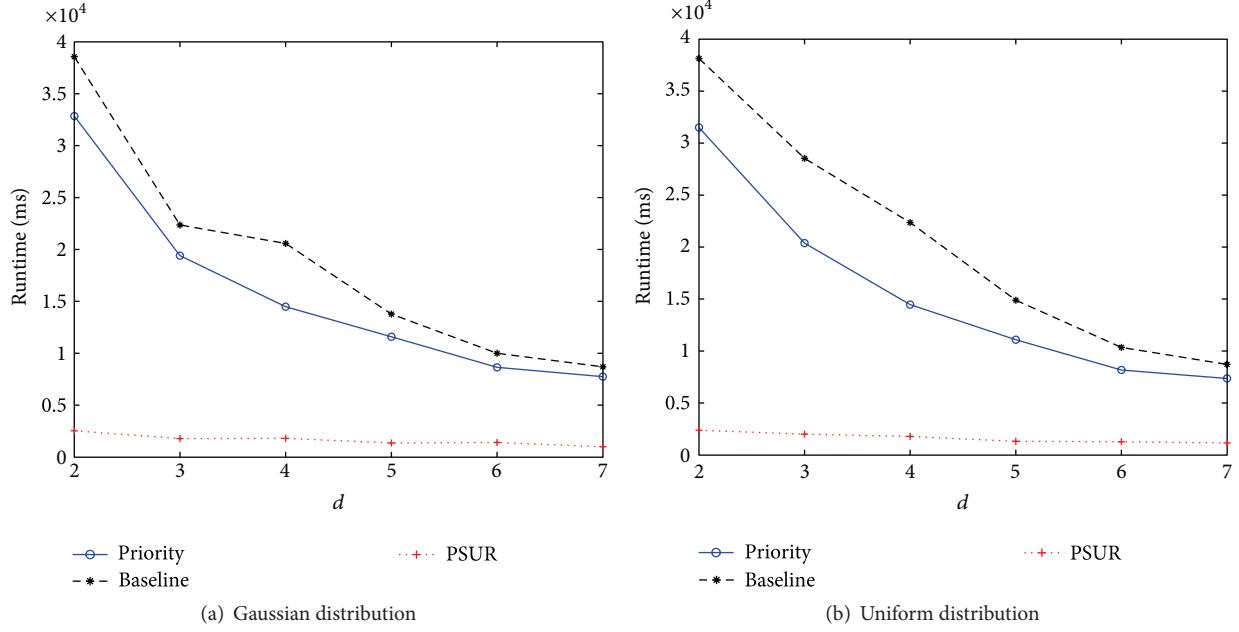


FIGURE 10: Static attribute dimensionality effects of the performance in Oldenburg city (Num = 50, Len = 20, $t = 20$).

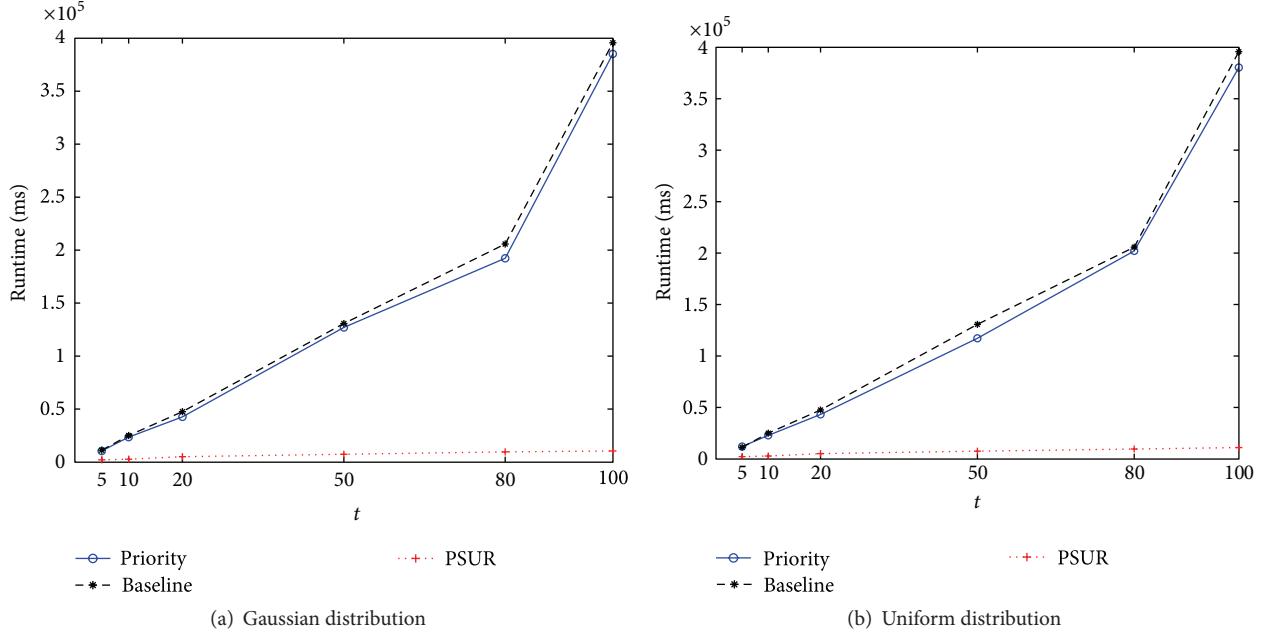


FIGURE 11: Time span effects of the performance in Cixi city ($d = 5$, Len = 20, Num = 100).

can update the p -skyline set by tracking event size to decide which one might vary, so the time cost is approximately the same as in each timestamp, which saves more time than other two algorithms. Figures 11 and 12 show this evidence.

Tables 8 and 9 show the event size changes with time span length of PSUR. It is obvious that the event size grows with increasing time span length. Nevertheless, the difference is not significant. The runtime of PSUR varies a little in Figures 11 and 12.

7. Conclusion

To the best of our knowledge, this is the first work to compute probabilistic skyline queries for uncertainty in road network. In this paper, we have addressed the problem of continuous probabilistic skyline query for moving objects. Firstly attributes of objects are divided into static and dynamic to define dominant probability and skyline probability with continuous data. In order to update the skyline

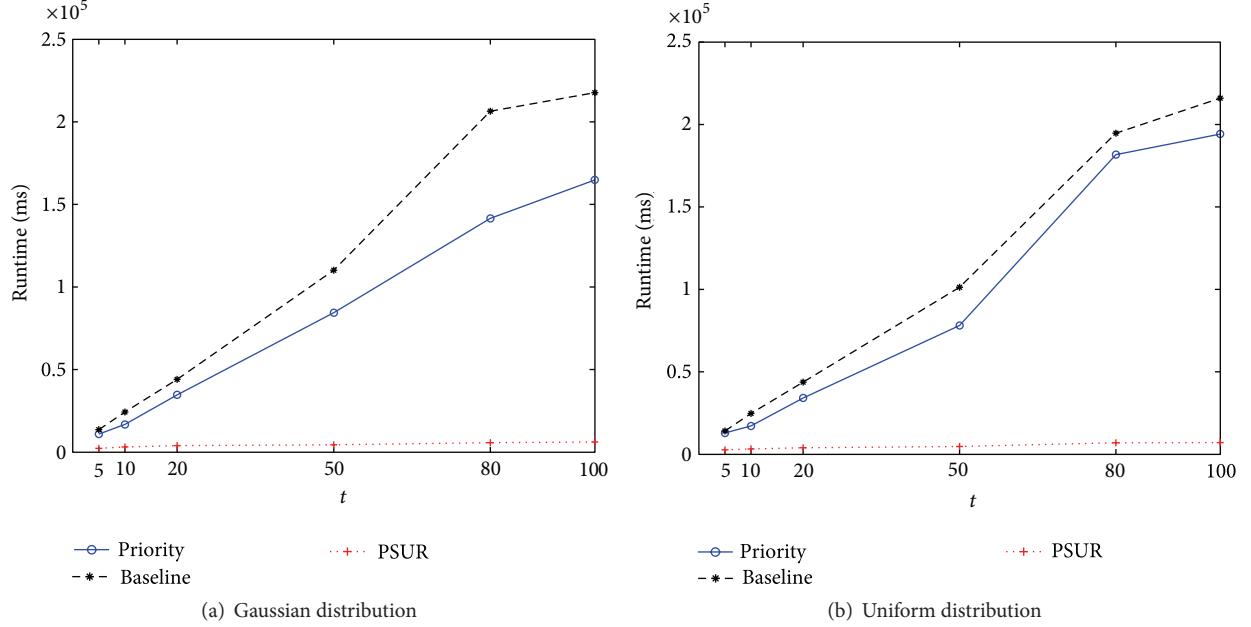


FIGURE 12: Time span effects of the performance in Oldenburg city ($d = 5$, Len = 20, Num = 100).

TABLE 4: Event size in Cixi road work ($d = 5$; Num = 50; $t = 20$).

Segment length	Event size before pruning	Event size after pruning
2	1029	39
10	1597	51
20	2264	82
40	3387	113
60	4147	137
100	4450	144

TABLE 5: Event size in Oldenburg road work ($d = 5$; Num = 50; $t = 20$).

Segment length	Event size before pruning	Event size after pruning
2	190	2
10	245	5
20	306	6
40	461	7
60	611	7
100	890	16

set continuously, trigger events are introduced to compute varying skyline probability of objects. Two pruning ways are proposed to save search space and speed up this computation. At last, the continuous probabilistic skyline query algorithm with uncertainty in road network named PSUR is proposed. Finally, a series of experiments are devised to verify the effectiveness and efficiency of PSUR. The results of our

TABLE 6: Event size in Cixi road network (Len = 20; Num = 50; $t = 20$).

Static dimensionality d	Event size before pruning	Event size after pruning
2	543	97
3	558	72
4	665	37
5	519	21
6	620	10
7	625	5

TABLE 7: Event size in Oldenburg road network (Len = 20; Num = 50; $t = 20$).

Segment length	Event size before pruning	Event size after pruning
2	101	41
10	118	14
20	121	7
40	124	6
60	128	4
100	122	2

experimental study for different scales of datasets on two real road networks are very encouraging. The investigation demonstrates that the proposed updating method is more effective and efficient than periodic methods.

TABLE 8: Event size in Cixi road network (Len = 20; Num = 100; $d = 5$).

Time span length	Event size before pruning	Event size after pruning
5	1029	39
10	1597	51
20	2264	82
50	3387	113
80	4147	137
100	4450	144

TABLE 9: Event size in Oldenburg road network (Len = 20; Num = 100; $d = 5$).

Segment length	Event size before pruning	Event size after pruning
5	359	9
10	481	13
20	583	15
50	714	20
80	830	24
100	914	28

Conflict of Interests

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

Acknowledgments

This research is partly supported by the Zhejiang Natural Science Foundation (LY12F02020), Ningbo Natural Science Foundation (2013A610063, 2012A610066), and the K. C. Wong Magna Fund in Ningbo University.

References

- [1] O. Wolfson, A. P. Sistla, S. Chamberlain, and Y. Yesha, “Updating and querying databases that track mobile units,” *Distributed and Parallel Databases*, vol. 7, no. 3, pp. 257–287, 1999.
- [2] G. S. Iwerks, H. Samet, and K. Smith, “Continuous k-nearest neighbor queries for continuously moving points with updates,” in *Proceedings of the International Conference on Very Large Data Bases (VLDB '03)*, pp. 512–523, 2003.
- [3] H. Mokhtar, J. Su, and O. Ibarra, “On moving object queries,” in *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS '02)*, pp. 188–198, June 2002.
- [4] Z. Huang, H. Lu, B. C. Ooi, and A. K. H. Tung, “Continuous skyline queries for moving objects,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 12, pp. 1645–1658, 2006.
- [5] R. Cheng, D. V. Kalashnikov, and S. Prabhakar, “Querying imprecise data in moving object environments,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1112–1127, 2004.
- [6] D. Pfoser and C. Jensen, “Capturing the uncertainty of moving-objects representations,” in *Proceedings of International Conference on Scientific and Statistical Database Management*, vol. 1651 of *Lecture Notes in Computer Science*, pp. 111–131, 1999.
- [7] X. Ding and Y. Lu, “Indexing the imprecise positions of moving objects,” in *Proceedings of the ACM SIGMOD Ph.D. Workshop on Innovative Database Research*, pp. 45–50, Beijing, China, 2007.
- [8] C. Böhm, F. Fiedler, A. Oswald, C. Plant, and B. Wackersreuther, “Probabilistic skyline queries,” in *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM '09)*, pp. 651–660, Hong Kong, November 2009.
- [9] S. Borzsonyi, D. Kossmann, and K. Stocker, “The skyline operator,” in *Proceedings of the 17th International Conference on Data Engineering*, pp. 421–430, Heidelberg, Germany, April 2001.
- [10] J. Chomicki, P. Godfrey, and J. Gryz, “Skyline with presorting: theory and optimization,” in *Proceedings of the 14th International Conference on Intelligent Information System*, pp. 593–602, 2005.
- [11] D. Kossmann, F. Ramsak, and S. Rost, “Shooting stars in the sky: an online algorithm for skyline queries,” in *Proceedings of the International Conference on Very Large Data Bases (VLDB '02)*, pp. 275–286, Hong Kong, 2002.
- [12] D. Papadias, Y. Tao, G. Fu, and B. Seeger, “Progressive skyline computation in database systems,” *ACM Transactions on Database Systems*, vol. 30, no. 1, pp. 41–82, 2005.
- [13] Y. Tao and D. Papadias, “Maintaining sliding window skylines on data streams,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 3, pp. 377–391, 2006.
- [14] L. Tian, P. Zou, A.-P. Li, and Y. Jia, “Grid index based algorithm for continuous skyline computation,” *Chinese Journal of Computers*, vol. 31, no. 6, pp. 998–1012, 2008.
- [15] M. Kontaki, A. N. Papadopoulos, and Y. Manolopoulos, “Continuous k-dominant skyline computation on multidimensional data streams,” in *Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08)*, pp. 956–960, March 2008.
- [16] Y. Ishikawa, Y. Iijima, and J. X. Yu, “Spatial range querying for Gaussian-based imprecise query objects,” in *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE '09)*, pp. 676–687, April 2009.
- [17] F. Fiedler, *Probabilistic skyline queries on uncertain data [Ph. D. dissertation]*, 2006.
- [18] J. Pei, B. Jiang, X. Lin et al., “Probabilistic skylines on uncertain data,” in *Proceedings of the 33th International Conference on Very Large Databases(VLDB '07)*, pp. 253–264, Vienna, Austria, 2007.
- [19] X. Lian and L. Chen, “Monochromatic and bichromatic reverse skyline search over uncertain databases,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*, pp. 213–226, June 2008.
- [20] W. Zhang, X. Lin, Y. Zhang, W. Wang, and J. X. Yu, “Probabilistic skyline operator over sliding windows,” in *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE '09)*, pp. 1060–1071, April 2009.
- [21] M. J. Atallah and Y. Qi, “Computing all skyline probabilities for uncertain data,” in *Proceedings of the 28th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS '09)*, pp. 279–287, July 2009.
- [22] X. Ding, X. Lian, L. Chen, and H. Jin, “Continuous monitoring of skylines over uncertain data streams,” *Information Sciences*, vol. 184, no. 1, pp. 196–214, 2012.

- [23] X. Lin, J. Xu, and H. Hu, “Range-based skyline queries in mobile environments,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 4, pp. 835–849, 2013.
- [24] X. Huang and C. S. Jensen, “In-route skyline querying for location-based services,” in *Proceedings of the 4th International Workshop on Web and Wireless Geographical Information Systems (W2GIS '04)*, vol. 3428 of *Lecture Notes in Computer Science*, pp. 120–135, November 2004.
- [25] K. Deng, X. Zhou, and H. T. Shen, “Multi-source skyline query processing in road networks,” in *Proceedings of the 23rd International Conference on Data Engineering (ICDE '07)*, pp. 796–805, April 2007.
- [26] S. M. Jang and J. S. Yoo, “Processing continuous skyline queries in road networks,” in *Proceedings of the International Symposium on Computer Science and Its Applications (CSA '08)*, pp. 353–356, October 2008.
- [27] H.-P. Kriegel, M. Renz, and M. Schubert, “Route skyline queries: a multi-preference path planning approach,” in *Proceedings of the 26th IEEE International Conference on Data Engineering (ICDE '10)*, pp. 261–272, March 2010.
- [28] W. Lee, C. S.-H. Eom, and T.-C. Jo, “Path skyline for moving objects,” *Web Technologies and Applications*, Springer, vol. 7235, pp. 610–617, 2012.

Research Article

An Efficient Confidentiality and Integrity Preserving Aggregation Protocol in Wireless Sensor Networks

Liehuang Zhu,¹ Zhen Yang,¹ Jingfeng Xue,² and Cong Guo¹

¹ Beijing Engineering Research Center of Massive Language Information Processing and Cloud Computing Application, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China

² School of Software, Beijing Institute of Technology, Beijing 100081, China

Correspondence should be addressed to Jingfeng Xue; xuejf@bit.edu.cn

Received 24 December 2013; Accepted 7 January 2014; Published 13 February 2014

Academic Editor: Zhongwen Guo

Copyright © 2014 Liehuang Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks (WSNs) are composed of sensor nodes with limited energy which is difficult to replenish. In-network data aggregation is the main solution to minimize energy consumption and maximize network lifetime by reducing communication overhead. However, performing data aggregation while preserving data confidentiality and integrity is challenging, because adversaries can eavesdrop and modify the aggregation results easily by compromised aggregation nodes. In this paper, we propose an efficient confidentiality and integrity preserving aggregation protocol (ECIPAP) based on homomorphic encryption and result-checking mechanism. We also implement ECIPAP on SimpleWSN nodes running TinyOS. Security and performance analysis show that our protocol is quite efficient while persevering both aggregation confidentiality and integrity.

1. Introduction

Wireless sensor networks consist of a large number of sensor nodes which can be deployed to sense, transmit, and process the data collected from the environment in many applications such as battlefield surveillance, health care monitoring, and traffic regulation. The environment data will be transmitted to the base station (BS) hop by hop. Sensor nodes have limited energy storage and their computational ability is not so powerful as base station. We cannot supply energy to the nodes at all times because they are usually deployed in the areas humans hardly reach. If some key nodes cannot work as we supposed, the whole network will break down. The energy of the node is mainly spent on data transmission. On TelosB nodes, sending and receiving one bit data cost $0.72 \mu\text{J}$ and $0.81 \mu\text{J}$ which are more than 600 times than 1.2nJ cost on processing one instruction by processor [1]. Therefore, how to reduce communication overhead is the key issue to prolong the life of wireless sensor networks [2, 3].

Other than transmitting data directly to the base station, wireless sensor nodes can perform aggregation operations,

such as SUM and AVERAGE, on aggregation nodes. Through this approach, the communication overhead will be reduced largely. An aggregation tree should be built before data aggregation. The topology can be tree-based or cluster-based depending on the applications. Then the sensors need to collect environment data from the monitoring area and transmit the data to the bases station hop by hop. When the intermediate nodes receive the data from their child nodes, they aggregate them using aggregation functions.

Sensor nodes should be deployed in the hostile areas in some applications, such as battlefield surveillance and target tracking. If these data was leaked to the enemies, it would bring heavy losses. The adversaries can contact the physical nodes easily and they can tamper the original sensing data so that the base station will receive the false aggregation results. If the base station cannot receive the true data, it will make wrong decisions. Some other attacks also affect the aggregation process like denial of service (DoS) or data selective forwarding. General data aggregation protocols do not consider these attacks, so they are useless in military applications. Designing a data aggregation protocol

should take security into consideration. All above, secure data aggregation mandates three security factors.

1.1. Data Confidentiality. The original sensing data and the intermediate aggregation results should not be disclosed to the unauthorized parties during the transmission process.

1.2. Data Integrity. If the original sensing data or the intermediate aggregation results were tampered by the adversaries or modified due to the poor quality of communications, the base station can detect the changes from final aggregation results.

1.3. Data Freshness. Sensing data received by the base station should represent the very recent state of the environment. If the aggregation values are too old, base station should discard such results.

Researchers have proposed many data aggregation protocols which can provide some secure protections. Most protocols can only provide confidentiality protection [4, 5] or integrity protection [6, 7]. Some protocols claim that they can protect both data confidentiality and data integrity, but some problems still exist. Data must be transmitted in some special forms if data integrity needs to be protected. But data form may be changed when some encryption algorithms are used to protect data confidentiality. They are contradictory in a way. Designing a data aggregation protocol which can provide both data confidentiality and data integrity is a challenge.

In this paper, we propose an efficient confidentiality and integrity preserving aggregation protocol (ECIPAP) in wireless sensor networks motivated by protocols SIES [8] and SHIA [9]. Our scheme is based on a lightweight homomorphic encryption which is energy efficient. Through the result-checking phase, every node in the wireless sensor network can verify if its data was added to the final aggregation results. We use a random number dissemination mechanism to update the keys stored in sensor nodes so as to guarantee the data freshness against replay attacks. This protocol also uses μ TESLA [10] to broadcast authenticated queries along with the random numbers. We implement our protocol on TelosB physical nodes. Through the results of the experiment and the theoretic analysis, we will show the practicability and high efficiency of our protocol.

The paper is organized as follows. In Section 2, we introduce the related work in this research area. Section 3 explains the system model in our protocol. Section 4 describes our protocol in detail. Section 5 analyses the security, experiment, and performance of ECIPAP. We conclude this paper in Section 6.

2. Related Work

Hu and Evan proposed the first integrity-preserving hierarchy data aggregation protocol in 2003 [11]. The main idea of this protocol is delayed aggregation and delayed authentication. Girao et al. proposed an aggregation scheme that can guarantee the end-to-end data confidentiality using symmetric key based homomorphic encryption [12]. After that, a series of secure data aggregation protocols have been

proposed. Existing protocols, which claim that they can protect both data confidentiality and data integrity, have some potential risks. Generally, we can think that both confidentiality and integrity preserving protocols should satisfy the following conditions.

- (i) The encrypted sensing data should only be decrypted at the base station but not on the intermediate nodes thus to implement end-to-end confidentiality.
- (ii) If the adversaries tamper the sensing data, base station can verify the data integrity through checking the final aggregation results.
- (iii) The energy consumption should be reasonable because of the limited energy in each node.

The protocols proposed in [13–16] use pair keys to encrypt the data transmitted between two nodes based on the existing integrity-preserving data aggregation protocols. These protocols achieve hop by hop confidentiality but not end-to-end confidentiality. Based on a confidentiality protection mechanism, paper [17] provides data integrity protection through using a global key $k^{(r)}$ and a key $k^{(r)l}$ shared with the base station to compute the authenticated message $m_i k^{(r)} + k^{(r)l}$ of message m_i . This protocol is not secure if an adversary compromises a node and gets the global key $k^{(r)}$, using which the adversary can forge message as wished. Protocol in [18] uses EC-ElGamal to achieve confidentiality protection. It also uses a global key k and the node's private key z_i to compute the signature of message m_i . However, through compromising a sensor node the adversary can get k which can be used to tamper the cipher and compute the corresponding signature. Paper [19] proposed a protocol that can protect the data confidentiality. Meanwhile, this protocol also can protect the aggregation data sent by cluster header from being tampered. But adversary can compromise a node and get the public key of the cluster; then the aggregation data can be tampered freely.

Chan et al. presented a provable secure tree-based in-network data aggregation protocol named SHIA [9]. Without assuming a particular data structure, SHIA can detect any manipulation of the in-network aggregation. SHIA has three phases: query dissemination phase, aggregation-commit phase, and result-checking phase. After the queries were broadcast to the whole network, sensor nodes send their environment data and commitment upward. In the result-checking phase, associated off-path values would be passed down to the aggregation tree. Through this way, sensor nodes can verify if their data was indeed added to the final aggregation results. After node i is convinced, it will send an authentication message $\text{MAC}(K_i, N \parallel \text{OK})$, where OK is a unique identifier and K_i is the key shared with base station, to base station hop by hop. The intermediate nodes perform XOR on these authentication messages and the final result received by base station is $\text{MAC}(K_1, N \parallel \text{OK}) \oplus \dots \oplus \text{MAC}(K_n, N \parallel \text{OK})$. Base station computes this message itself and compares it with the authentication message received. If two messages match, base station accepts the aggregation result. Otherwise, the result will be ignored.

Although this protocol can protect data integrity, the data confidentiality is not guaranteed.

Combining homomorphic encryption and secret sharing, Papadopoulos et al. proposed a secure data aggregation protocol SIES which can protect both data confidentiality and data integrity [8]. In this protocol, each sensor node has a secret message ss_i , a global key K , and a private key k_i shared with base station. After the environment data v_i was sensed from monitoring area, each sensor node generates a message $m_i = v_i \parallel ss_i$. Sensor nodes use homomorphic encryption to encrypt m_i and send them to their parent nodes. The parent nodes can perform aggregation functions on the encrypted data directly. When the final aggregation result is received by the base station, it will be decrypted and divided into two parts: environment data V and secret SS. Base station computes the SS itself and compares it with the secret received. This protocol also has a problem that if the data v_i of the message m_i was tampered by adversaries, base station cannot detect such changes.

Our protocol improves SIES [8] by using result-checking mechanism motivated by SHIA [9] to protect data integrity instead of secret sharing. We also use homomorphic encryption to allow intermediate nodes perform aggregation directly so as to achieve end-to-end data confidentiality. When the final aggregating result was sent to the base station, the result-checking process will detect whether all the data is added to the final aggregation result.

3. System Model

Different secure data aggregation protocols support different aggregation functions. The security protection abilities are also various and the adversaries can launch a variety of attacks. This section gives the general problem definition, network assumption, and attack model in detail.

3.1. Network Assumption. We assume that an aggregation tree is already set up in the deployment phase. If not ready, TAG [20] can be used to build such tree-based networks. In our protocol, base station BS has unlimited energy and computing ability. BS can broadcast query messages to the whole network using authenticated method μ TESLA [10]. In the aggregation tree, there are two types of nodes, leaf nodes and the intermediate nodes. Leaf nodes only sense environment data and encrypt them, while intermediate nodes not only sense environment data but also aggregate their child nodes' data. The distance between two nodes is about 10 m and nodes can only communicate with its child nodes or parent nodes.

Each sensor node has a unique ID and an initial key k_i . We further assume that sensor nodes can perform symmetric additively homomorphic encryption and collision-resistant hash function.

3.2. Problem Definition. In our protocol, we use an efficient symmetric additively homomorphic encryption algorithm. A key distribution mechanism is also used to generate the keys in each sensor node.

3.2.1. Homomorphic Encryption. As shown in Algorithm 1, M is a large integer. Let $m(m \in [0, M])$ be the message that needs to be encrypted, and $k(k \in [0, M])$ is the initial key stored in every node before deployment. Parameter r is the random number sent by base station and H is the hash function.

3.2.2. Message Format. The message m sent by sensor node has a fixed format which is a data tuple:

$$\langle \text{count}, \text{value}, \overline{\text{value}}, \text{MAC} \rangle, \quad (1)$$

where count is the number of the sensor nodes in the subtree. If a sensor node is the leaf node, then count = 1. Parament value is the environment data and it has lower bound V_{\min} and upper bound V_{\max} . We also add $\overline{\text{value}} = V_{\max} - \text{value}$ to this data tuple. MAC can be computed as follows:

$$\text{MAC} = H(\text{count} \parallel \text{value} \parallel \overline{\text{value}} \parallel \text{ID}_i), \quad (2)$$

where ID_i is node's unique ID.

3.2.3. Aggregation Function. An intermediate node i has n child nodes s_1, s_2, \dots, s_n . Let v_1, v_2, \dots, v_n be the data sensed by these child nodes from the monitory area. We define aggregation function F_{agg} that can be performed on intermediate nodes as shown in Algorithm 2.

3.3. Attack Model. The most threatening attack against data aggregation in wireless sensor networks is node compromising launched by the adversaries. Through this attack, adversaries can obtain the keys of the nodes. After that, they can use fake messages to disturb the final aggregation results. Such stealthy attacks [12] can make base station accept the false aggregation results without being detected. We assume that the adversaries can compromise a fraction of sensor nodes in the network.

Denial-of-service attack is out of our consideration, because this type of attack can be detected easily by the base station when the network works improperly.

4. ECIPAP

We improve SIES [8] by using result-checking mechanism instead of secret sharing. Then we propose an efficient confidentiality and integrity preserving aggregation protocol (ECIPAP). The homomorphic encryption used in our protocol can guarantee end-to-end data confidentiality. Our protocol has three phases: query dissemination phase, data aggregation phase, and result-checking phase.

4.1. Network Deployment. Before the sensor nodes were deployed in the monitory area, each sensor node shared a private key k_i , a large integer M , and a unique ID_i with the base station. The symmetric additively homomorphic encryption algorithm and hash function SHA-1 are also preset. When the aggregation process begins, we assume that the aggregation tree is already set up. We can use TAG [20] to form an aggregation tree.

(i) Key Generation: Round key $k_r = H(k, r) \bmod M$
(ii) Encryption: Ciphertext $c = \text{Enc}(m, k_r, M)$ $= (m + k_r) \bmod M$
(iii) Decryption: $m = \text{Dec}(c, k_r, M) = (c - k_r) \bmod M$
(iv) Addition: $c_i = \text{Enc}(m_i, k_{i,r}, M)$ $c_j = \text{Enc}(m_j, k_{j,r}, M)$ $c = (c_i + c_j) \bmod M$ $m_i + m_j = \text{Dec}(c_i + c_j, k_{i,r} + k_{j,r}, M)$

ALGORITHM 1: Homomorphic encryption algorithm.

(i) Count Function: $F_{\text{count}} = \sum_{i=1}^n m_i \cdot \text{count}$
(ii) Sum Function: $F_{\text{sum}} = \langle \sum_{i=1}^n m_i \cdot \text{value}, \sum_{i=1}^n m_i \cdot \overline{\text{value}} \rangle$
(iii) Average Function: $F_{\text{average}} = \left\langle \frac{\sum_{i=1}^n m_i \cdot \text{value}}{\sum_{i=1}^n m_i \cdot \text{count}}, \frac{\sum_{i=1}^n m_i \cdot \overline{\text{value}}}{\sum_{i=1}^n m_i \cdot \text{count}} \right\rangle$
(iv) MAC Aggregation: $F_{\text{MAC}} = \bigoplus_{i=1}^n m_i \cdot \text{MAC}$

ALGORITHM 2: Aggregation functions.

4.2. Query Dissemination Phase. In each aggregation round, base station BS chooses a random number r and the query message such as COUNT, SUM, and AVERAGE. Due to its powerful communication ability, it can broadcast this query message to the whole network along with the random number r . When the nodes receive the query messages, they store them in the RAMs and start the data aggregation phase.

4.3. Data Aggregation Phase. Sensor node collects environment data value $\in [V_{\min}, V_{\max}]$ such as temperature and sets $\text{value} = V_{\max} - v$, $\text{count} = 1$. Next it generates the temporary keys $k_{i,r}$, $k'_{i,r}$, and $k''_{i,r}$:

$$\begin{aligned} k_{i,r} &= H(k_i, r) \bmod M, \\ k'_{i,r} &= H(k_i, r+1) \bmod M, \\ k''_{i,r} &= H(k_i, r+2) \bmod M. \end{aligned} \quad (3)$$

Then it encrypts these data as follows:

$$\begin{aligned} c_{\text{count},i} &= \text{Enc}(\text{count}, k_{i,r}, M), \\ c_{\text{value},i} &= \text{Enc}(\text{value}, k'_{i,r}, M), \\ c_{\overline{\text{value}},i} &= \text{Enc}(\overline{\text{value}}, k''_{i,r}, M). \end{aligned} \quad (4)$$

The reason why we use different keys to encrypt count, value, and $\overline{\text{value}}$ will be shown latter. Sensor nodes can compute message authentication code as follows:

$$\text{MAC}_i = H(c_{\text{count},i} \parallel c_{\text{value},i} \parallel c_{\overline{\text{value}},i} \parallel \text{ID}_i). \quad (5)$$

The data tuple can be created now as

$$u_i = \langle c_{\text{count},i}, c_{\text{value},i}, c_{\overline{\text{value}},i}, \text{MAC}_i \rangle. \quad (6)$$

When the nodes prepare the data tuples ready, they send them to their parent nodes. We show the aggregation process in Figure 1. There are $N = 10$ nodes in the network and node F has two child nodes I and J . When node F receives the data tuples u_i and u_j sent by node I and node J , it aggregates them with the data tuple u_f created by itself: $U_f = F_{\text{agg}}(u_i, u_j, u_f)$. Node F sends this intermediate aggregation result to its parent node C . The final aggregation result will be transmitted to the base station BS at last. BS can decrypt this aggregation result as follows:

$$\begin{aligned} \text{count}_{\text{agg}} &= \text{Dec}\left(U_a \cdot \text{count}, \sum_{i=1}^n k_{i,r}, M\right), \\ \text{value}_{\text{agg}} &= \text{Dec}\left(U_a \cdot \text{value}, \sum_{i=1}^n k'_{i,r}, M\right), \\ \overline{\text{value}}_{\text{agg}} &= \text{Dec}\left(U_a \cdot \overline{\text{value}}, \sum_{i=1}^n k''_{i,r}, M\right). \end{aligned} \quad (7)$$

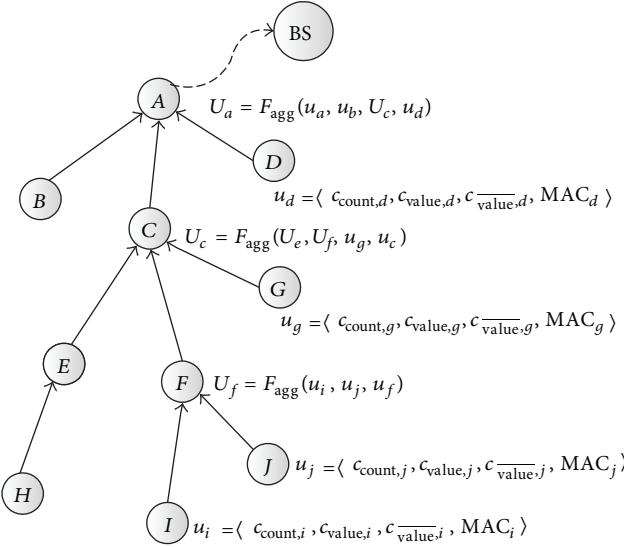


FIGURE 1: Data aggregation phase in ECIPAP.

The base station BS can also get the MAC_{agg} :

$$\text{MAC}_{\text{agg}} = \bigoplus_{i=1}^n m_i \cdot \text{MAC}. \quad (8)$$

4.4. Result-Checking Phase. When the final aggregation result is received, base station broadcasts the aggregated data tuple down to the whole network using authenticated method. To enable result checking, each sensor node will send a checking message to its child node. We can regard the result checking as a reverse process of data aggregation. We show this in Figure 2. Base station sends the final data tuple U'_a to node A. Then node A can remove the data sent by nodes B, C, and D from U'_a . We define the reverse computational operation as Θ . For example, we can get data tuple created by node A as $u'_a = U'_a \Theta u_b \Theta U_c \Theta u_d$. Through comparing u'_a with the data tuple u_a created by node A, node A can verify if its data was added to the final aggregation result.

Every sensor node can verify if its own data was added to the aggregation data by comparing its own data to the data sent by parent nodes. If the result passes the verification, then every sensor node prepares an authentication message $\text{MAC}(k_i \parallel r \parallel \text{OK})$ and sends it to the base station hop by hop. If verification is failed, $\text{MAC}(k_i \parallel r \parallel \text{NO})$ will be sent. When the intermediate sensor nodes receive these authentication messages, they aggregate them using MAC Aggregation Function. The base station also can calculate this authentication message with its own data stored before network deployment. So comparing these two authentication messages can verify if all the sensing data is added to the final aggregation result. If the authentication message passes the verification, then base station accepts this aggregation result. Otherwise, base station just ignores it.

5. Analysis

5.1. Security Analysis. Adversaries can compromise a fraction of sensor nodes in the wireless sensor network. When

TABLE 1: Characteristics of simpleWSN nodes.

CPU 8-bit	8 MHz
Storage	10 Kbytes RAM
	48 Kbytes FLASH
Communication	2.4 GHz
Bandwidth	250 Kbps
Operating system	TinyOS 2.1

TABLE 2: Parameters used in experiment.

N	5	Total node number
r	$[1, 2^{16} - 1]$	Random number
$k_{i,r}, k'_{i,r}, k''_{i,r}$	$[0, 2^{32} - 1]$	Encryption keys
count	$[0, 2^8 - 1]$	Aggregating count
value	$[0, 2^{16} - 1]$	Aggregating value
$\overline{\text{value}}$	$[0, 2^{16} - 1]$	$V_{\max} - \text{value}$
MAC_i	160 bit	SHA-1 output
M	2^{32}	Large integer
V_{\min}	1000	Lower bound
V_{\max}	4000	Upper bound
OK/NO	1/0	Identification
t	1000 ms	Query period

a sensor node is compromised, its private information such as encryption keys will be leaked. Adversaries can sniff confidential data sent by the sensor nodes. They can launch stealthy attack to make the base station accept false data without being detected.

We assume the adversaries can eavesdrop messages sent between sensor nodes and they already know the range of sensing data. If we use the same key to encrypt both value and $\overline{\text{value}}$, the adversaries can get the original sensing data through algebraic operation, such as $\text{value}_i = (\text{value}_{i,i} - \overline{\text{value}}_{i,i} + V_{\max})/2$. Because we use different keys $k_{i,r}$, $k'_{i,r}$, and $k''_{i,r}$ to encrypt count, value, and $\overline{\text{value}}$ in ECIPAP, data confidentiality can be protected.

Authors in [9] define the *direct data injection* and *optimally secure* as follows.

5.1.1. Direct Data Injection. A *direct data injection* attack occurs when an adversary modifies the data readings reported by the nodes under its direct control, under the constraint that only legal readings in $[V_{\min}, V_{\max}]$ are reported.

5.1.2. Optimally Secure. An aggregation algorithm is *optimally secure* if, by tampering with the aggregation process, an adversary is unable to induce the base station to accept any aggregation result which is not already achievable by *direct data injection*.

The message format in our protocol is the same as the message format $\langle \text{count}, \text{value}, \text{complement}, \text{commitment} \rangle$ used in SHIA [9]. Adversaries can compromise some nodes and set all their readings to V_{\min} or V_{\max} so that the final aggregation result is in the range $[\mu V_{\min}, \mu V_{\max}]$ where μ is the number of malicious nodes. Obviously, any aggregation

TABLE 3: Encrypted aggregation results received by BS.

count	value	value	MAC
8A 9D A4 86	20 25 79 BE	16 09 0C C5	21 B9 95 E0 86 36 3D 1E F5 9C 06 12 70 BE B1 53 05 AE 81 D4
08 EF 90 17	04 98 E7 85	61 03 3E 1B	76 4A 9B 43 A9 74 5D CB 70 30 D5 D9 43 6E E9 93 29 1D 1B 2C
B9 84 48 53	25 0E 1A AE	6E AD 98 D8	DA 31 9E CA F6 04 A5 1F 21 4C 80 77 CF 75 E7 2B 9C 5A EB 65
A1 1E 78 01	02 A8 59 FD	AA 50 50 0E	35 E2 76 0A 08 EE DB 1E 80 D5 44 77 4A 78 8A 59 CE D9 A8 2F
E0 B1 ED 95	1F 13 C7 A4	B8 53 B3 3E	1B 49 94 BE B3 A0 0F B3 7D 27 56 33 77 11 B8 BE 48 87 34 4F
0D 15 51 8D	04 16 05 14	C8 66 EF C8	1E A4 2C 8B 99 1B 70 E2 BD C1 4C 6F 1B A5 84 34 0C 50 31 7A
36 A5 79 3F	0C CF 54 28	A4 21 21 74	EC 51 67 47 9E CA 90 FA 64 DF 80 25 19 C1 10 01 45 F7 9A 14

TABLE 4: The original environment data received by BS after decryption.

count	value	value	MAC
00 00 00 05	00 00 0B 55	00 00 15 77	74 21 49 83 DA ED D4 83 97 21 FA E7 AD 88 9F 58 FC C9 78 AA
00 00 00 05	00 00 0B 54	00 00 15 79	9C A6 75 EF 33 0C 88 9A 90 93 57 C4 73 12 F0 D0 52 6D 39 CE
00 00 00 05	00 00 0B 55	00 00 15 77	F1 37 76 9A 0F 43 66 CA 86 B0 92 34 CC 9A 75 50 25 E4 93 1C
00 00 00 05	00 00 0B 56	00 00 15 72	86 37 5C B3 CB EB C3 B6 2C 92 72 56 B7 46 A9 7C CD 76 6E 00
00 00 00 05	00 00 0B 55	00 00 15 75	DE B3 93 F6 98 E6 D1 E8 D9 D6 35 73 DD D1 34 B5 62 D7 8A A8
00 00 00 05	00 00 0B 55	00 00 15 76	5C AF 15 67 B6 28 5E F6 2F 5A A1 06 89 09 E3 D2 B6 CC C5 06
00 00 00 05	00 00 0B 55	00 00 15 75	98 1D 7E 47 B2 49 B0 96 DE B7 F1 BD EA 00 E0 39 6B AE CB E5

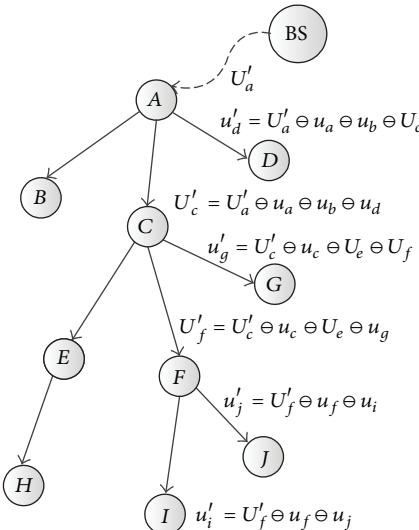


FIGURE 2: Result-checking phase in ECIPAP.

result between these two bounds can be achieved by direct data injection. Hence, our ECIPAP is optimally secure.

5.2. Experiment Analysis. We have implemented ECIPAP using SimpleWSN experimental platform to sense average temperature in lab. Characteristics of SimpleWSN nodes are shown in Table 1.

The temperature data DataNum sensed by SimpleWSN nodes are 16-bit hexadecimal integers. We can use

the following formula to transform such data to readable Celsius degree:

$$\text{Temperature (C)} = \frac{((\text{DataNum}/4096) * 1.5 - 0.98)}{0.00355}. \quad (9)$$

Table 2 describes the parameters used in our experiment. We use sensor nodes to form an aggregation tree whose root is a powerful PC.

Table 3 shows the data received by base station before decryption. The data in count, value, and value columns are encrypted. We know this environment data is larger than the upper bound V_{\max} , so we cannot get any useful information from such messages.

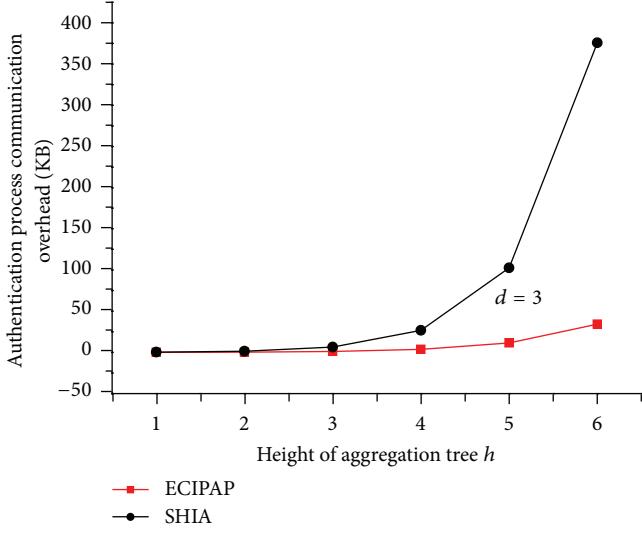
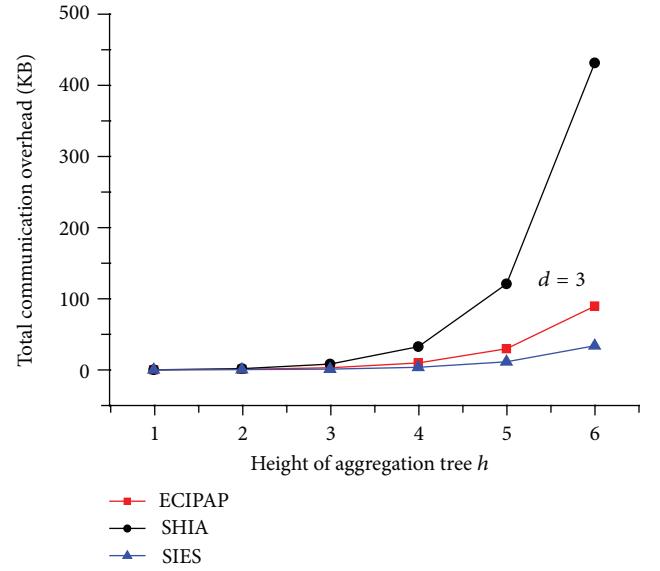
Then, we use the keys shared between sensor nodes and base station to decrypt the aggregation results and show them in Table 4. The total number of nodes used is 5 and the average temperature is about 23.2 degrees Celsius. Base station broadcasts query messages every 1000 million seconds. Our experiment shows the practicability and high efficiency of ECIPAP.

5.3. Performance Analysis. We give a theoretical analysis of the performance in this section. Through comparing the communication overhead between ECIPAP, SHIA, and SIES, we show that ECIPAP not only protect both data integrity and data confidentiality but also reduce the communication overhead in result-checking phase.

We assume every intermediate node in the network has d child nodes and the height of the aggregation tree is h . The length of the data tuple sent by sensor nodes in ECIPAP, SHIA, and SIES are $|m|$, $|m'|$ and $|m''|$. The length of the

TABLE 5: Communication overhead of ECIPAP, SHIA, and SIES.

	ECIPAP	SHIA	SIES
Edge congestion	$ m $	$h \times (d - 1) \times m' $	$ m'' $
Node congestion	$d \times m $	$d \times h \times (d - 1) \times m' $	$d \times m'' $
Data aggregation	$\sum_{i=1}^h d^i \times m $	$\sum_{i=1}^h d^i \times m' $	$\sum_{i=1}^h d^i \times m'' $
Authentication	$\sum_{i=1}^h d^i \times m $	$\sum_{i=1}^h d^i \times (d - 1) \times i \times m' $	—
MAC transmission	$\sum_{i=1}^h d^i \times \text{MAC} $	$\sum_{i=1}^h d^i \times \text{MAC}' $	—

FIGURE 3: Communication overhead in authentication process when node degree $d = 3$.FIGURE 4: Total communication overhead in ECIPAP, SHIA, and SIES when node degree $d = 3$.

authentication messages, sent by sensor nodes in result-checking phase, are $|\text{MAC}|$ and $|\text{MAC}'|$ in ECIPAP and SHIA. The number of nodes in the whole network is $\sum_{i=1}^h d^i$. So we can get the communication overhead of ECIPAP, SHIA, and SIES in Table 5.

In ECIPAP, we use 4-byte integers to present encrypted count, value, and value. Hash function SHA-1 outputs a 160-bit value. So we know $|m| = 256$ bit and $|\text{MAC}| = 160$ bit. Message length and MAC length in SHIA are the same as ECIPAP, so we can set $|m'| = 256$ bit and $|\text{MAC}'| = 160$ bit. The length of message in SIES is also 256 bit. The communication overhead in SHIA is mostly spent in authentication process. Our protocol can reduce such overhead largely as we present in Figure 3.

The total communication overhead reduces by 79% comparing with SHIA when $h = 6$ and $d = 3$, shown in Figure 4. As the height of aggregation tree increases, the advantages of ECIPAP are more obvious. ECIPAP improves the secure ability of SIES with small communication overhead increase. Our protocol can work more efficiently in the applications where data should be transmitted along multihops.

6. Conclusion

It is difficult to design an aggregation protocol to protect both data confidentiality and data integrity in wireless sensor

networks. Existing protocols which claim to achieve this goal still have some problems. In this paper, we improve SIES by using result-checking mechanism instead of secret sharing. We also use homomorphic encryption algorithm to protect data end-to-end confidentiality. We concentrate on the communication overhead cost in the result-checking phase and propose an efficient confidentiality and integrity preserving aggregation protocol (ECIPAP). We implement our protocol on physical nodes running TinyOS. Through this experiment and theoretical analysis, we show the practicability and high efficiency of ECIPAP.

Conflict of Interests

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

Acknowledgments

This paper is supported by National Natural Science Foundation of China (no. 61272512, no. 61003262, and no. 61300177), Program for New Century Excellent Talents in University (NCET-12-0047), and Beijing Natural Science Foundation (no. 4121001, no. 4132054).

References

- [1] M. M. Groat, W. B. He, and S. Forrest, "KIPDA: k-indistinguishable privacy-preserving data aggregation in wireless sensor networks," in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2024–2032, Shanghai, China, April 2011.
- [2] M. Li and Y. H. Liu, "Iso-map: energy-efficient contour mapping in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 5, pp. 699–710, 2010.
- [3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [4] T. Feng, C. Wang, W. Zhang, and L. Ruan, "Confidentiality protection for distributed sensor data aggregation," in *Proceedings of the 27th IEEE International Conference on Computer Communications (INFOCOM '08)*, pp. 68–76, Phoenix, Ariz, USA, 2008.
- [5] E. Mykletun, J. Girao, and D. Westhoff, "Public key based cryptoschemes for data concealment in wireless sensor networks," in *Proceedings of the 4th IEEE International Conference on Communications (ICC '06)*, vol. 5, pp. 2288–2295, Istanbul, Turkey, July 2006.
- [6] M. Manulis and J. Schwenk, "Security model and framework for information aggregation in sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, no. 2, pp. 1–28, 2009.
- [7] L. Yu, J. Li, S. Cheng, and S. Xiong, "Secure continuous aggregation via sampling-based verification in wireless sensor networks," in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 1763–1771, Shanghai, China, 2011.
- [8] S. Papadopoulos, A. Kiayias, and D. Papadias, "Exact in-network aggregation with integrity and confidentiality," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 10, pp. 1760–1773, 2012.
- [9] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06)*, pp. 278–287, Alexandria, Va, USA, November 2006.
- [10] A. Perrig, R. Szewczyk, V. Wen, J. D. Tygar, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.
- [11] L. Hu and D. Evan, "Secure aggregation for wireless networks," in *Proceedings of the IEEE Symposium on Applications and the Internet Workshops (SAINT '03)*, vol. 2003, pp. 384–391, Orlando, Fla, USA.
- [12] J. Girao, D. Westhoff, and M. Schneider, "CDA: concealed data aggregation for reverse multicast traffic in wireless sensor networks," in *Proceedings of the 40th IEEE International Conference on Communications (ICC '05)*, pp. 3044–3049, Seoul, Republic of Korea, May 2005.
- [13] J. Pawan and M. Anish, "Efficient secure aggregation in sensor networks," in *Proceedings of the 11st International Conference on High-Performance Computing (HiPC '04)*, pp. 40–49, 2004.
- [14] E.-O. Blaß, J. Wilke, and M. Zitterbart, "Relaxed authenticity for data aggregation in wireless sensor networks," in *Proceedings of the 4th International Conference on Security and Privacy in Communication Networks (SecureComm '08)*, pp. 1–10, Istanbul, Turkey, September 2008.
- [15] S. Ozdemir and H. Çam, "Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 736–749, 2010.
- [16] M. S. Hung, C.-H. Chen, and P.-C. Li, "A lightweight secure data aggregation protocol for wireless sensor networks," in *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW '11)*, pp. 101–107, Taipei, Taiwan, September 2011.
- [17] C. Castelluccia, A. C.-F. Chan, E. Mykletun, and G. Tsudik, "Efficient and provably secure aggregation of encrypted data in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, no. 3, pp. 1–36, 2009.
- [18] J. Albathe and S. Madria, "Secure hierarchical data aggregation in wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '09)*, pp. 1–6, Budapest, Hungary, April 2009.
- [19] S. Ozdemir and Y. Xiao, "Integrity protecting hierarchical concealed data aggregation for wireless sensor networks," *Computer Networks*, vol. 55, no. 8, pp. 1735–1746, 2011.
- [20] S. Madden, J. M. Franklin, J. Hellerstein et al., "TAG: a tiny aggregation service for adhoc sensor networks," in *Proceedings of the 5th Annual Symposium on Operating Systems Design and Implementation (OSDI '02)*, pp. 131–146, Boston, Mass, USA, 2002.