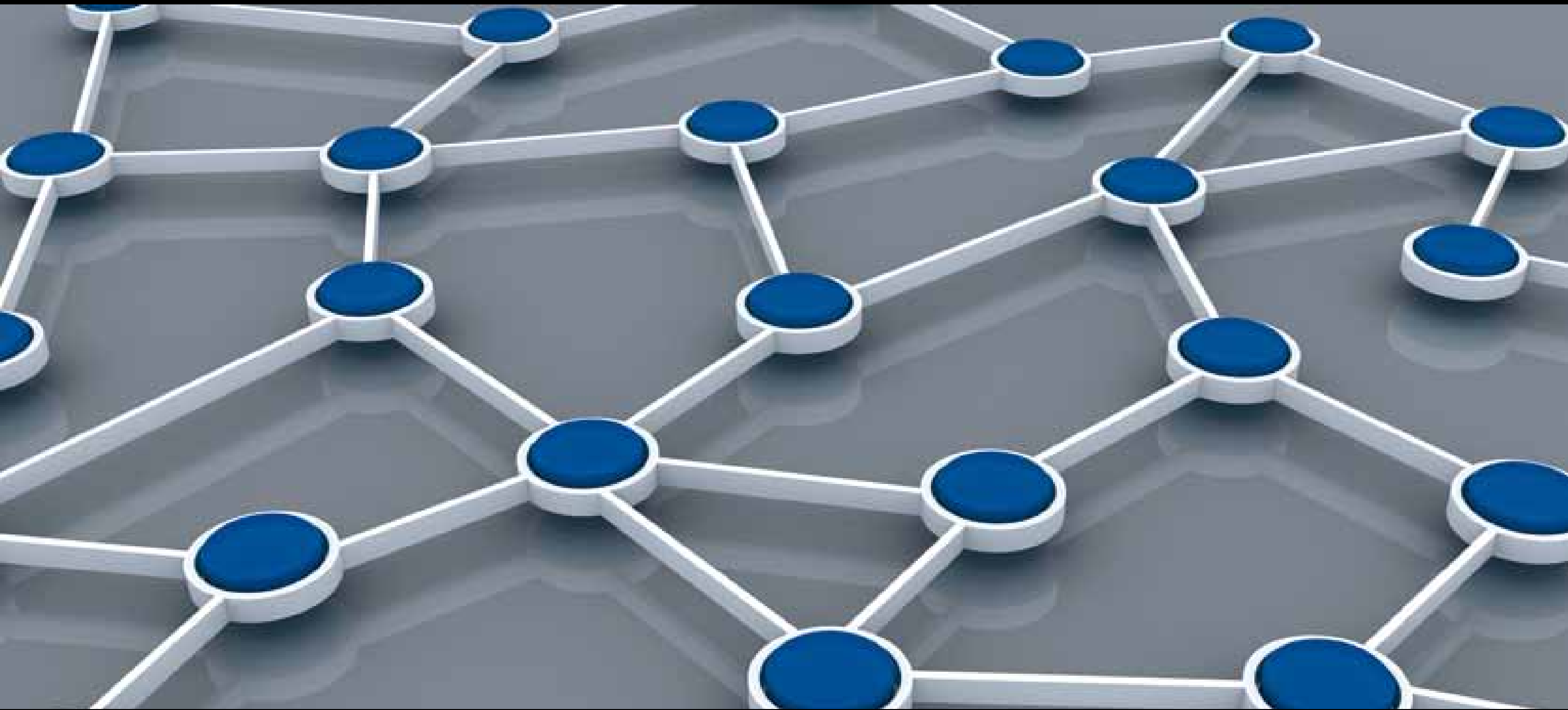
International Journal of Distributed Sensor Networks

Secure and Energy-Efficient Data Collection in Wireless Sensor Networks

Guest Editors: Anfeng Liu, Laurence T. Yang, Motoki Sakai, and Mianxiong Dong



Secure and Energy-Efficient Data Collection in Wireless Sensor Networks

International Journal of Distributed Sensor Networks

Secure and Energy-Efficient Data Collection in Wireless Sensor Networks

Guest Editors: Anfeng Liu, Laurence T. Yang, Motoki Sakai, and Mianxiong Dong

Editorial Board

Contents

Secure and Energy-Efficient Data Collection in Wireless Sensor Networks, Anfeng Liu, Laurence T. Yang, Motoki Sakai, and Mianxiong Dong
Volume 2013, Article ID 565076, 3 pages

A Secure and Efficient Data Aggregation Framework in Vehicular Sensing Networks, Suguo Du, Peng Tian, Kaoru Ota, and Haojin Zhu
Volume 2013, Article ID 298059, 10 pages

An Improved Trust Model Based on Interactive Ant Algorithms and Its Applications in Wireless Sensor Networks, Yun Pan, Yafang Yu, and Li Yan
Volume 2013, Article ID 764064, 9 pages

Trust Management Scheme Based on D-S Evidence Theory for Wireless Sensor Networks, Renjian Feng, Shenyun Che, Xiao Wang, and Ning Yu
Volume 2013, Article ID 948641, 9 pages

Low Mismatch Key Agreement Based on Wavelet-Transform Trend and Fuzzy Vault in Body Area Network, Yang Wu, Yongmei Sun, Lei Zhan, and Yuefeng Ji
Volume 2013, Article ID 912873, 16 pages

An Energy-Efficient Key Predistribution Scheme for Secure Wireless Sensor Networks Using Eigenvector, Sung Jin Choi, Kyung Tae Kim, and Hee Yong Youn
Volume 2013, Article ID 216754, 13 pages

Real-Time Seismic Data Acquisition via a Paired Ripple Transmission Protocol, Jin-Ling Lin, Kao-Shing Hwang, Hui-Kai Su, and Min-Che Hsieh
Volume 2013, Article ID 765973, 9 pages

Efficient and Secure Routing Protocol Based on Encryption and Authentication for Wireless Sensor Networks, Jiliang Zhou
Volume 2013, Article ID 108968, 17 pages

A Hybrid Authenticated Group Key Agreement Protocol in Wireless Sensor Networks, Yue Li, Dehua Chen, Wei Li, Gaoli Wang, and Paul Smith
Volume 2013, Article ID 716265, 14 pages

Constructing a CDS-Based Network Backbone for Data Collection in Wireless Sensor Networks, Xiaoyan Kui, Yu Sheng, Huakun Du, and Junbin Liang
Volume 2013, Article ID 258081, 12 pages

An Energy-Efficient Cyclic Diversionary Routing Strategy against Global Eavesdroppers in Wireless Sensor Networks, Ju Ren, Yaoxue Zhang, and Kang Liu
Volume 2013, Article ID 834245, 15 pages

Energy-Efficiency of Cooperative Communication with Guaranteed E2E Reliability in WSNs, Deyu Zhang and Zhigang Chen
Volume 2013, Article ID 532826, 11 pages

Noncommutative Lightweight Signcryption for Wireless Sensor Networks, Lize Gu, Yun Pan, Mianxiong Dong, and Kaoru Ota
Volume 2013, Article ID 818917, 10 pages

A Cross-Layer Security Scheme of Web-Services-Based Communications for IEEE 1451 Sensor and Actuator Networks, Jun Wu, Ming Zhan, Bin Duan, and Jiang Liu
Volume 2013, Article ID 138384, 10 pages

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 565076, 3 pages
http://dx.doi.org/10.1155/2013/565076

Editorial

Secure and Energy-Efficient Data Collection in Wireless Sensor Networks

Anfeng Liu,[1] Laurence T. Yang,[2] Motoki Sakai,[3] and Mianxiong Dong[4]

[1] *School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2] *Department of Computer Science, St. Francis Xavier University, Antigonish, NS, Canada B2G 2W5*
[3] *School of Symbiotic Systems Science and Technology, Fukushima University, Fukushima 960-1296, Japan*
[4] *School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu 965-8580, Japan*

Correspondence should be addressed to Anfeng Liu; afengliu@mail.csu.edu.cn

Received 12 August 2013; Accepted 12 August 2013

Wireless sensor networks (WSNs) are a kind of multihop and self-organizing networks formed by the way of wireless communication and composed by a large number of low-cost microsensor nodes deployed in the monitoring area. The sensor nodes cooperate to perceive and acquire the process and transmit the perceived object information in the network covering within the geographical area and finally send the information to the sink. In the recent decade, with the rapid development of various kinds of key technology in WSNs, they are widely applied to many especial environments, such as, military defense, environment detection, biological health, and disaster-relief work. Due to the limited computing ability, battery capacity, and storage capacity of sensor nodes, energy-efficient data collection became a non-negligible research issue in WSNs. Also the open architecture in receiving data and transmitting data of WSNs is vulnerable to various security attacks during the data collection process. These two fundamental issues motivate us to have this special issue addressing the recent advances which are mentioned in Table 1 on security and energy-efficient data collection in WSNs.

In response to this call for papers, we have received a total of 36 high-quality submissions, and 13 papers have been selected for publication after a rigorous review process due to the space limit. The papers in this special issue are divided into three thematic groups as follows.

The first set of the seven papers emphasizes the security technology with a series of key agreement protocol, encryption, authentication, and trust scheme to solve the special security issues. According to the current researches in security technology in Table 1, the seven papers are significant for the improvement of the security of the network. "*A hybrid authenticated group key agreement protocol in wireless sensor networks*" by Y. Li et al. proposed an AGKA protocol which reduced the high cost public-key operations at the sensor side and replaced them with efficient symmetric-key based operations. In "*Efficient and secure routing protocol based on encryption and authentication for wireless sensor networks*" by J. Zhou, the author aimed to perform a secure routing protocol based on encryption and authentication which encrypted all communication packets and authenticated the source nodes and the BS. "*Trust management scheme based on D-S evidence theory for wireless sensor network*" by R. Feng et al. proposed a trust management scheme based on revised Dempster-Shafer (D-S) evidence theory. "*An improved trust model based on interactive ant algorithms and its applications in wireless sensor networks*" by Y. Pan et al. improved the Marmol et al.'s ant algorithm based trust model. "*Low mismatch key agreement based on wavelet-transform trend and fuzzy vault in body area network*" by Y. Wu et al. introduces the security on the body area network with lightweight symmetric cryptography. "*Noncommutative lightweight signcryption for wireless sensor networks*" by L. Gu et al. has proposed a braid-based signcryption scheme and developed a key establishment protocol for wireless sensor networks. "*A cross-layer security scheme of web services-based communications for IEEE 1451 sensor and actuator networks*" by J. Wu et al. dealt with the requirements, of authentication, integrity, confidentiality and availability.

Table 1: The main content of secure and energy-efficient data collection researches in WSNs.

The main research fields	The main content in current years
key management, encryption, and authentication	Hierarchical key management of sensor network [1], lightweight, and strong security key agreement [2].
Secure routing	Multipath security routing protocol [3], hierarchical sensor network security routing protocol [4].
Secure data aggregation	Syntactic aggregation and cryptographic aggregation [5]; use the watermark rather than PH data fusion technology to achieve security [6].
Secure localization	Hierarchical sensor network security positioning [7], lightweight security localization method [8].
Privacy protection	Data oriented privacy protection, contextual privacy [9], source-location privacy, footprint privacy, and communication privacy [10].
Energy-efficiency routing	Hierarchy and clustering routing [11].
Energy-efficiency MAC protocol	The MAC protocol based on CSMA/FDMA/CSMA [12].

The second set of the three papers focuses on how to improve the network's energy efficiency and prolong the lifetime of the network within the security assurance. "*An energy-efficient key predistribution scheme for secure wireless sensor networks using eigenvector*" by S. J. Choi et al. proposes a new robust key predistribution scheme reducing the overhead requirement of secure connectivity and energy consumption. In "*An energy-efficient cyclic diversionary routing strategy against global eavesdroppers in wireless sensor networks*" by J. Ren et al. proposed an energy-efficient cyclic diversionary routing (CDR) scheme against global eavesdroppers for preserving location privacy and maximizing lifetime of wireless sensor networks. "*Energy-efficiency of cooperative communication with guaranteed E2E reliability in WSNs*" by D. Zhang and Z. Chen addressed the energy efficiency of cooperative communication in WSNs.

The last set of the rest mainly aims at the issues or the way of data collection. According to Table 1, the researches about secure data aggregation start in "*A secure and efficient data aggregation framework in vehicular sensing networks*" by S. Du et al. which introduced a basic aggregation scheme which could aggregate the data and the message authentication codes by using syntactic aggregation and cryptographic aggregation. A work that used a low-cost, reliable, and microchip-based wireless transmission solution to real-time collect earthquake data across local and wide areas is in "*Real-time seismic data acquisition via a paired ripple transmission protocol*" by J.-L. Lin et al. Besides, "*Constructing a CDS-based network backbone for data collection in wireless sensor networks*" by X. Kui et al. investigates the problem of constructing an energy balanced CDS to effectively preserve the energy of nodes in order to extend the network lifetime in data collection.

Compared with the recent researches in Table 1, all 13 papers in this special issue represent the spirit of innovation and important leaps in this field. They improve the performance in security and energy efficiency in data collection in WSNs and could be a guide and cornerstone to the future work.

Acknowledgments

The guest editor thanks all the authors who have contributed their time and effort submitting their outstanding works in response to the call for papers, regardless of whether their works have been published in this special issue or not on account of the space limitations. The guest editors are also grateful to all the reviewers for their valuable suggestions to the authors on improving the issues and presentation of their papers.

Anfeng Liu
Laurence T. Yang
Motoki Sakai
Mianxiong Dong

References

[1] S. Guo, A.-N. Shen, and M. Guo, "A secure and scalable rekeying mechanism for hierarchical wireless sensor networks," *IEICE Transactions on Information and Systems*, vol. 93, no. 3, pp. 421–429, 2010.

[2] L. Ni, G. Chen, and J. Li, "Escrowable identity-based authenticated key agreement protocol with strong security," *Computers & Mathematics with Applications*, vol. 65, no. 9, pp. 1339–1349, 2013.

[3] H. Yin, Y. Wang, G. Min, S. Berton, R. Guo, and C. Lin, "A secure multipath routing protocol in mobile *ad hoc* networks," *Concurrency and Computation*, vol. 22, no. 4, pp. 481–502, 2010.

[4] F. Bao, I.-R. Chen, M. Chang, and J.-H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection," *IEEE Transactions on Network and Service Management*, vol. 9, no. 2, pp. 169–183, 2012.

[5] C. Rottondi, G. Verticale, and C. Krauss, "Distributed privacy-preserving aggregation of metering data in smart grids," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1342–1354, 2013.

[6] S. Xi and X. Di, "A reversible watermarking authentication scheme for wireless sensor networks," *Information Sciences*, vol. 240, pp. 173–183, 2013.

[7] R. Garg, A. L. Varna, and M. Wu, "An efficient gradient descent approach to secure localization in resource constrained wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 717–730, 2012.

[8] K.-J. Kim and S.-P. Hong, "Privacy care architecture in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 369502, 7 pages, 2013.

[9] H. J. G. Oberholzer and M. S. Olivier, "Privacy contracts incorporated in a privacy protection framework," *Computer Systems Science and Engineering*, vol. 21, no. 1, pp. 5–16, 2006.

[10] Z. Zhu and G. Cao, "Toward privacy preserving and collusion resistance in a location proof updating system," *IEEE Transactions on Mobile Computing*, vol. 12, no. 1, pp. 51–64, 2013.

[11] H. T. Xiao, X. Zhao, and H. Ogai, "A new clustering routing algorithm for wsn based on brief artificial fish-school optimization and ant colony optimization," *IEEE Transactions on Electronics, Information and Systems*, vol. 133, no. 7, pp. 1339–1349, 2013.

[12] S. M. Kumar, N. Rajkumar, and W. C. C. Mary, "Dropping false packet to increase the network lifetime of wireless sensor network using EFDD protocol," *Wireless Personal Communications*, vol. 70, no. 4, pp. 1697–1709, 2013.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 298059, 10 pages
http://dx.doi.org/10.1155/2013/298059

Research Article

A Secure and Efficient Data Aggregation Framework in Vehicular Sensing Networks

Suguo Du,[1] Peng Tian,[1] Kaoru Ota,[2] and Haojin Zhu[1]

[1] *Shanghai Jiao Tong University, Shanghai 200240, China*
[2] *Muroran Institute of Technology, Muroran 050-8585, Japan*

Correspondence should be addressed to Haojin Zhu; zhu-hj@cs.sjtu.edu.cn

Received 4 February 2013; Accepted 18 March 2013

Academic Editor: Mianxiong Dong

Vehicular ad hoc networks support a wide range of promising applications including vehicular sensing networks, which enable vehicles to cooperatively collect and transmit the aggregated traffic data for the purpose of traffic monitoring. The reported literatures mainly focus on how to achieve the data aggregation in dynamic vehicular environment, while the security issue especially on the authenticity and integrity of aggregation results receives less attention. In this study, we introduce a basic aggregation scheme which could aggregate the data and the message authentication codes by using *syntactic aggregation* and *cryptographic aggregation*. To tolerate duplicate messages and further improve the aggregation performance, we introduce a secure probabilistic data aggregation scheme based on Flajolet-Martin sketch and *sketch proof* technique. We also discuss the tradeoff between the bandwidth efficiency and the estimation accuracy. Extensive simulations and analysis demonstrate the efficiency and effectiveness of the proposed scheme.

1. Introduction

With the advancement of wireless technology, vehicular communication networks, also known as vehicular ad hoc networks (VANETs), are emerging as a promising approach to increase road safety, efficiency, and convenience [1, 2]. Although the primary purpose of vehicular networks is to enable communication-based automotive safety applications, VANETs also allow a wide range of promising applications such as traffic monitoring and data collecting, which are regarded as an important component of future intelligent transportation systems (ITSs). It is also observed that rising popularity of smartphones with onboard sensors (e.g., GPS, compass, accelerometer) and always-on mobile Internet connections sheds light on using smartphones as a platform for large-scale vehicular sensing. Recent reports report that smartphone users have surpassed feature phone users in the USA by 2012. According to figures released by IDC, 207.6 million Android and Apple smart-phones were shipped in the fourth quarter of 2012. This further renders the possibility of vehicular sensing.

As shown in [3–10], Departments of Transportation in the USA must collect various types of data (e.g., average speed or traffic density) for traffic monitoring purposes. Traditionally, these important data are collected by technologies such as inductive loop detectors (ILDs), video detection systems, acoustic tracking systems, or microwave radar sensors. However, these technologies mostly suffer from a high maintenance cost. On the other hand, cooperative data collection and dissemination in VANETs allow the traffic monitoring performed in a more cost-effective way [11]. Specifically, each vehicle collects its own or neighboring information (e.g., its current speed or neighboring traffic) and then transmits it to the remote roadside units (RSUs) via vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. The RSUs can be deployed at various points of interest along the roadway and can be used to collect data from locations up to tens of kilometers away. In this study, we coin the vehicular networks which are designed for traffic sensing and monitoring as the *vehicular sensing networks*.

One of the major challenges of vehicular sensing networks is high overhead of transmitted sensing data. Each sensing result is essentially some spatial-temporal measured values (speed, traffic density), which record the position of vehicles (i.e., a road segment or a small area) and the observation time. Such sensing data is periodically broadcasted. Upon reception of such a broadcast, the intermediate receivers/forwarders incorporate the received data into their local reports and then broadcast them again. Unfortunately, such a periodical broadcast brings on a high traffic load or even *traffic storm*. This problem is more serious in the scenario of high vehicle density, which could be found on multilaned highways in congestion situations. On the other hand, in most cases, drivers or monitors do not need exact individual reports, but only an overview of the general average speed on the road ahead [12]. This motivates the data aggregation issues in vehicular networks, including Flajolet-Martin sketch based probabilistic aggregation [13], fuzzy aggregation [12], and others [14, 15]. However, most of them are mainly focusing on how to achieve the data aggregation in dynamic vehicular environment, while the security issues on the aspect of the authenticity and integrity of aggregation results receive less attention. Since aggregation operation could be made by any intermediate forwarding vehicle, any malicious attacker could easily launch the attacks towards the data aggregation process by modifying the aggregated result or simply inserting invalid sensing data.

Secure data aggregation is a great challenge in vehicular sensing networks due to their unique network characteristics including highly dynamic network topology, intermittent connectivity, and potentially huge numbers of VANET nodes. These unique characteristics make the secure data aggregation in traditional wireless sensor networks such as [16], which always assume either a static network topology or aggregation structure, unsuitable for vehicular sensing networks.

Therefore, to achieve secure and efficient data sensing and collection, in this paper, we present the SAS, a secure data aggregation scheme for vehicular sensing networks which includes the basic scheme and advanced scheme. In the basic scheme, it achieves efficient data and MAC authentication via syntactic aggregation and cryptographic aggregation. However, the basic scheme needs to keep the original sensing data, which prevents a more efficient data aggregation. Further, it cannot work in case of the existence of duplicate messages. Thus, to overcome this problem, we propose an advanced scheme based on Flajolet-Martin sketch and a series of sketch proof techniques. We also discuss the tradeoff between the bandwidth efficiency and the estimation precision. Finally, extensive simulations and analysis demonstrate the efficiency and effectiveness of the proposed scheme.

The remainder of the paper is organized as follows. In Section 2, we introduce the related work. In Section 3, we present the system model and the design goals. In Section 4, we present some preliminaries. In Section 5, we present a secure data aggregation scheme in vehicular sensing networks by using the syntactic aggregation and cryptographic aggregation approach. In Section 6, we propose a probabilistic data aggregation scheme. Performance analysis is given in Section 7, followed by the conclusion in Section 8.

2. Related Work

Vehicular sensing networks represent a promising way to cooperatively collect useful information in order to increase road safety and driver convenience for future intelligent transportation system. By being integrated with the traditional digital map system, vehicular sensing networks provide the functionality of real-time automatic route scheduling [14], decentralized free parking places discovery [15], traffic monitoring [3], and so forth. In these applications, data aggregation is necessary for efficient data propagation and reduced transmission overhead.

There are quite a few research proposals for data aggregation in vehicular sensing networks [14, 15]. Most of them are based on group formulation and vehicle clustering, which can dramatically reduce the communication overhead due to the increased aggregation level. In additional to the above proposals, the structure-free aggregation frameworks are also proposed including Flajolet-Martin sketch-based aggregation [13] and fuzzy aggregation [12] without defining aggregate structures. However, the aforementioned studies focus on the data aggregation itself but do not take the security issues into consideration.

The most related research study for secure data aggregation in VANETs is the voting scheme, including [17, 18], which involves multiple vehicles to collect information towards a specific event (e.g., collision or traffic jam). Each witness (or observer) of this event will submit a message to a group leader. The group leader will take the responsibility of collecting more than a threshold k of proofs from k distinct witnesses to prove the validity of an emergency event by the voting scheme. References [17, 18] discuss how to further improve the aggregation efficiency by exploiting cryptographic tools such as onion signature [18] and aggregate signature [17]. Note that, in this study, we consider a more general data aggregation scenario: collecting data within a certain area and, at the same time, providing security guarantee for the aggregation functionality.

3. System Model and Design Goal

This section describes our system model, attack model, security assumptions, and design goals.

3.1. Network Model. In this paper, we consider a general vehicular sensing network model, which is mainly comprised of three components: traffic monitoring centre (TMC), RSUs, and vehicles. As shown in Figure 1, RSUs could be selectively deployed at some positions (e.g., intersections) to collect the traffic information (e.g., average speed) within a certain area. Due to high maintenance cost, RSUs could be only deployed intermittently to reduce the deployment cost. We assume that each vehicle, which is equipped with an on-board unit (OBU), has the capability of data collecting and reporting. The transmitted sensing data are propagated via V2V and V2I communications to the RSUs, which then forward them to the TMC. SAS is based on the distributed aggregation model similar to [13], which does not require any group/cluster formulation.

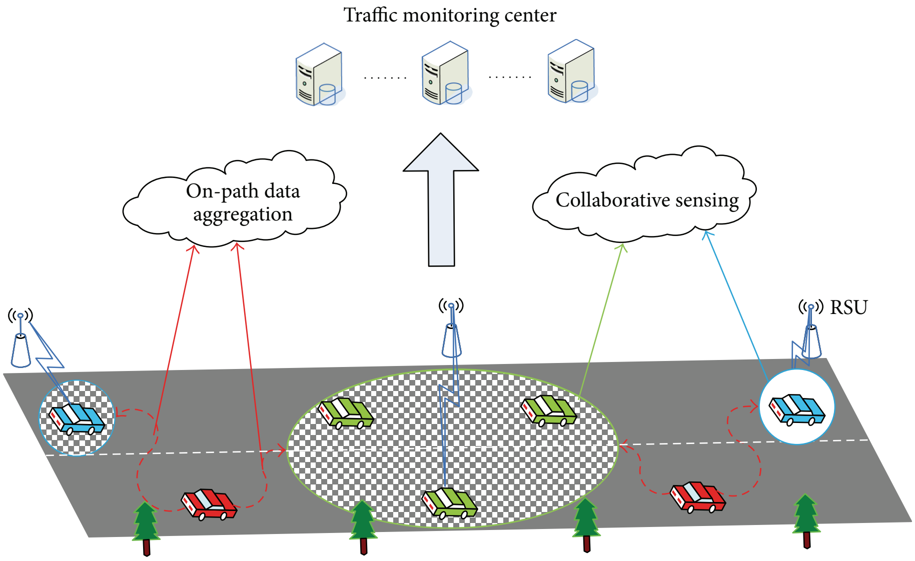


Figure 1: Overview of vehicular sensing network.

3.2. Security Assumptions. We assume that each OBU either shares a distinct secret symmetric key with TMC or obtains a public/private key pair, which is issued by TMC. Whether using shared secret key or public key depends on different system requirements.

3.3. Attack Model. In this study, we assume that the TMC and RSUs are trusted while vehicles (including the sensing vehicles and aggregator vehicles) are potentially malicious and can thus launch various attacks including fabricating, duplicating, and computing the aggregation incorrectly. We do not consider denial-of-service attacks where aggregator vehicles fail to or refuse to provide any acceptable result. A malicious sensor can always report an arbitrary sensing report, which fundamentally cannot be prevented. So we do not aim at preventing such an attack.

3.4. Design Goals

(i) *Security Goal.* The security goal of SAS is to enable the TMC to verify whether an aggregate sensing report is correct or not. Specifically, TMC should accept a reported aggregate report if and only if it is equal to the output of a correct execution of the aggregation function over all of the sensing reports provided by the qualified vehicles in the most recent epoch.

(ii) *Efficiency and Effectiveness Goal.* The efficiency goal of SAS is to minimize the transmission overhead and, at the same time, to ensure a certain sensing accuracy. However, computational cost is not a major concern of this paper since VANET is generally assumed to have unlimited computational capability [17].

4. Preliminaries

4.1. One-Way Chains and MAX Protocols. One-way chain is a widely used cryptographic primitive, which is based on a one-way function F and a secret seed s. The one-way function F is easy to compute but computationally infeasible to invert. The chain has the sequence of values $F(s)$, $F(F(s))$, $F(F(F(s)))$, ... Throughout this paper, we use $F^x()$ to denote recursively applying the function F for x times. Thus, the xth value of the sequence is $F^x(s)$. For example, given two positive integers m and n, where $m < n$, it is easy to compute the $F^n(s)$ by functioning forward the value of $F^m(s)$ for $(n-m)$ times with the function F. However, it is infeasible to compute the value of $F^m(s)$ by functioning backward the value of $F^n(s)$. One-way chain has been widely used in many security topics such as micropayment. Recently, in [16], the authors take advantage of one-way chains to construct a MAX protocol, which could ensure the aggregated maximum message cannot be inflated or deflated. However, MAX protocol is not designed for probabilistic aggregation. Further, the network topology considered in [16] is for sensor networks with statistic network topology. In SAS, what we consider is a dynamic network topology and probabilistic aggregation model.

4.2. Pairing Technique. The proposed basic scheme is based on bilinear pairing which is briefly introduced as below. Let $\mathbb{G}$ be a cyclic additive group and $\mathbb{G}_T$ a cyclic multiplicative group of the same prime-order q; that is, $|\mathbb{G}| = |\mathbb{G}_T| = q$. Let g be a generator of $\mathbb{G}$ and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ an efficient admissible bilinear map with the following properties:

(i) bilinear: for $a, b \in \mathbb{Z}_q^*$, $e(g^a, g^b) = e(g, g)^{ab}$;

(ii) nondegenerate: $e(g, g) \neq 1$.

4.3. Aggregate Signature and Batch Verification. The major computation cost for authenticating an emergency message comes from verifying a set of supporting signatures issued by different emergency witnesses. The corresponding public key certificates of the signers also need to be verified together. All of them will incur a significant amount of transmission and verification cost. In this study, we use aggregate signature to reduce the transmission cost of supporting signatures, certificates, and batch verification to realize efficient signature verification.

An aggregate signature is a digital signature that supports aggregation of n distinct signatures issued by n distinct signers to a single short signature [19]. This single signature (and the n original messages) will convince the verifier that the n signers indeed sign the n original messages. In addition to the benefit of the reduced transmission size, aggregate signature technique supports batch verification, which enables the receivers to quickly verify a set of digital signatures on different messages by different signers. In this study, we adopt the aggregate signature and batch verification introduced in [20] as our basic cryptographic aggregation technique to improve the aggregation performance.

5. A General Secure Data Aggregation Framework in Vehicular Sensing Networks

In this section, we introduce a general data aggregation framework in vehicular sensing networks by using the syntactic aggregation and cryptographic aggregation approach.

5.1. System Setup. The TMC generates a tuple $(q, g, \mathbb{G}, \mathbb{G}_T, e)$ as the system parameters. The TMC selects a random $sk \in \mathbb{Z}_q^*$ as its secret key and generates its public key $pk = g^{sk}$, by which four hash functions are formed: $H : \{0,1\}^* \rightarrow \mathbb{G}$, $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$, $H_2 : \{0,1\}^* \rightarrow \mathbb{G}$, $H_3 : \{0,1\}^* \rightarrow \mathbb{Z}_q$. The group public key and secret key are $(q, g, \mathbb{G}_1, \mathbb{G}_T, e, pk, H, H_1, H_2, H_3)$ and sk, respectively.

An important task of the setup procedure is to determine the format of emergency report message. In our study, the format of a secure sensing report (SSR) is defined as follows. For a sensed event, the sensor vehicle i will generate an SSR:

$$\mathrm{SSR}_i = \left(\mathrm{ID}_i, \mathrm{Type\#}, v_i, \mathrm{Loc\#}, \mathrm{epoch\#}, \mathrm{MAC}_i, \mathrm{Cert}_i\right), \quad (1)$$

where ID_i denotes the identity of the vehicle that generates the claim. Type# denotes the type of SSR reported in this report. v_i denotes the sensing value provided by i. Loc# denotes the sensing area. epoch# denotes the sensing period. MAC_i denotes the message authentication code generated by vehicle i on this SSR. It has two modes: symmetric key mode (Mode I) or public key mode (Mode II). Cert_i denotes the certificate held by vehicle i.

For a specific event, it is reasonable to assume that the relevant SSRs will share the same Type#, Loc#, and epoch#.

5.2. Registration. A vehicle can join the network by performing the following step depending on Mode I or Mode II.

(1) *Private Key Generation for Mode I.* In the symmetric key mode, a vehicle i can randomly choose x_i as its secret key.

(2) *Private/Public Key Generation for Mode II.* In the public key mode, a vehicle can randomly choose $x_i \in \mathbb{Z}_q^*$ as its secret key and generate its public key $X_j = g^{x_j}$. After ensuring the legitimacy of this vehicle, TMC will issue the public key certificate by signing its signature on (i, X_i). Here, the certificate generation process follows a typical Boneh, Lynn, and Shacham signature scheme in [19]. TMC computes $h_i \leftarrow H(i \parallel X_i)$ and $\sigma_i \leftarrow h_i^{x_i}$. $\mathrm{Cert}_i = (i, X_i, \sigma_i)$ is the public key certificate of i. The verification of public key certificate could be as follows. Given a vehicle's public key certificate Cert_i, $h_i \leftarrow H(i \parallel X_i)$ can be computed, and it is accepted if $e(\sigma_i, g) = e(h_i, pk)$.

5.3. SSR Generation and Broadcasting. Once an event is sensed by one or multiple vehicles and the observation is (Type#, Loc#, epoch#), the sensing vehicles $i \mid i = 1, 2, \ldots$ may independently generate their SSRs as follows.

(1) *Mode I SSR Generation.* In terms of Mode I SSR generation, given the type and observation time of the emergency message $\mathrm{TL} = \mathrm{Type\#} \parallel \mathrm{epoch\#}$ as well as the location information $\ell = \mathrm{Loc\#}$, a witness vehicle i with its private key x_i could compute message authentication code as follows:

$$\mathrm{MAC}_i = H\left(x_i, i, \mathrm{Type\#}, \mathrm{Loc\#}, \mathrm{epoch\#}\right). \quad (2)$$

Thus, $(i, \mathrm{Type\#}, \mathrm{Loc\#}, \mathrm{epoch\#}, \mathrm{MAC}_i)$ constitutes an SSR claim generated by vehicle i towards the sensing event. After that, i will broadcast this SSR to its neighbors.

(2) *Mode II SSR Generation.* For Mode II SSR, given the type and observation time of the emergency message $\mathrm{TL} = \mathrm{Type\#} \parallel \mathrm{epoch\#}$ as well as the location information $\ell = \mathrm{Loc\#}$, a witness vehicle with its public and private key pairs (X_j, x_j) can compute $w_i \leftarrow H_3(\mathrm{TL} \parallel \ell)$, $a \leftarrow H_1(\ell)$, $b \leftarrow H_2(\ell)$ and generate the signature $\mathrm{MAC}_i = a^{x_i} b^{x_i w_i}$. Thus, $(i, \mathrm{Type\#}, \mathrm{Loc\#}, \mathrm{epoch\#}, \mathrm{MAC}_i, \mathrm{Cert}_i)$ constitutes an SSR claim generated by vehicle i towards the sensing event. After that, i will broadcast this SSR to its neighbors.

A single SSR verification can be performed as follows: given $\mathrm{SSR} = (i, \mathrm{Type\#}, \mathrm{Loc\#}, \mathrm{epoch\#}, \mathrm{MAC}_i, \mathrm{Cert}_i)$, the verifier will first check the validity of certificate included in this SSR. After that, it can check the validity of supporting signature by computing $w_i \leftarrow H_3(\mathrm{TL} \parallel \ell)$, $a \leftarrow H_1(\ell)$, $b \leftarrow H_2(\ell)$. It is accepted if $\mathrm{MAC}_i = a^{x_i} b^{x_i w_i}$.

5.4. SSR Opportunistic Forwarding. In VANETs, the network topology could be very dynamic and diversified in shape from time to time, even sometimes sparse and frequently partitioned. The communication between vehicles is expected to be performed in an opportunistic manner. This means a vehicle can carry packets when routes do not exist but forward the packets to the new receivers when they move into its vicinity [21]. To enable the opportunistic data propagation, vehicles that are within a range r and maintain connectivity for a minimum time t with each other can be arranged to form a cluster. The detailed discussion on cluster creation and maintenance can be found in [21]. We refer to the node at the head of every cluster as header, which is responsible for forwarding the data to the next cluster in a typical opportunistic data forwarding algorithm such as [21, 22]. The messages will be buffered at the header until they are forwarded to the next cluster, which is also referred to as the "*Carry and forward*" strategy. In this study, it is considered that the header can also play the role of emergency message aggregator because of the following two reasons.

(1) If taking a header of a cluster as the aggregator, the aggregation process will be merged into a part of data forwarding process. Therefore, there is no need to elect another cluster head to perform the data aggregation operations.

(2) The process of message propagation between two clusters is referred to as a catch-up process, where a message traverses along with its carrying vehicles until it reaches within the radio range of the vehicle at the end of another cluster, which obviously presents a considerable propagation interval depending on the speed of vehicles and the gap between clusters. Therefore, we can use such an interval to aggregate the related emergency messages to minimize the aggregation latency.

In the following sections, a cluster head will be taken as the aggregator of the cluster, which will perform the following SSR aggregated authentication algorithm.

5.5. SSR Secure Aggregation. For any specific emergency event, each aggregator maintains two local message lists, which keep the forwarded SSRs and ReadytoForward SSRs, respectively. The forwarded message list, denoted as $\mathscr{F}$, contains all the SSRs which have been forwarded by this vehicle before, while the ReadytoForward message list, denoted as $\mathscr{R}$, stores messages which have not been transmitted but can be forwarded some time later. The SSRs set $\mathscr{F} \cup \mathscr{R}$ includes all the SSRs related to a specific event. Whenever receiving an SSR, the aggregator should check if this SSR is a duplicate. If yes, such an SSR will be dropped; otherwise it will be put into the message list R. Before the forwarded propagation, the aggregator will perform the SSR aggregation (or *Aggregate_SSR*) and SSR batch verification (*BatchVerify_SSR*) operations as follows.

5.5.1. SSR Aggregation. Aggregate_SSR is used to aggregate multiple SSRs into a single SSR, which includes two steps: *syntactic aggregation* step and *cryptographic aggregation* step.

(i) *Syntactic Aggregation*. For a specific event, given n SSRs $(i, \text{Type\#}, \text{Loc\#}, \text{epoch\#}, \text{MAC}_i, \text{Cert}_i)$ by vehicles $1, \ldots, n$, we can obtain syntactically aggregated SSR as $\text{SSR}_{\text{agg}} = (1, \ldots, n, \text{Type\#}, \text{Loc\#}, \text{epoch\#}, \text{MAC}_1, \ldots, \text{MAC}_n, \text{Cert}_1, \ldots, \text{Cert}_n)$.

(ii) *MAC Aggregation*. It is used to aggregate multiple MACs into a single MAC, which includes the following two modes: Mode I and Mode II.

(1) *Mode I Aggregation*. Mode I aggregation is

$$\begin{aligned}\text{MAC}_{\text{agg}} = {} & H(x_1, 1, \text{Type\#}, \text{Loc\#}, \text{epoch\#}) \\ & \otimes \cdots \otimes H(x_n, n, \text{Type\#}, \text{Loc\#}, \text{epoch\#}),\end{aligned} \tag{3}$$

where $\otimes$ can be XOR operation.

(2) *Mode II Aggregation*. Mode II aggregation includes the certificate aggregation $\text{Cert}_{\text{agg}} \leftarrow (i, X_i, \sigma_{\text{agg}})$ and MAC aggregation $\sigma_{\text{agg}} \leftarrow \prod_{i=1}^{n} \text{Cert}_i$. $\text{MAC}_{\text{agg}} \leftarrow \prod_{i=1}^{n} \text{MAC}_i$.

After syntactic aggregation and cryptographic aggregation, we can obtain the aggregated SER as $\text{SSR}_{\text{agg}} = (1, \ldots, n, \text{Type\#}, \text{Loc\#}, \text{epoch\#}, \text{MAC}_{\text{agg}}, \text{Cert}_{\text{agg}})$.

5.5.2. SSR Batch Verification. In this section, we exploit batch verification to further reduce the computational cost.

(i) *Mode I Verification*. For Mode I verification, TMC could verify the sensing reports by verifying the following equations:

$$\begin{aligned}\text{MAC}_{\text{agg}} = {} & H(x_1, 1, \text{Type\#}, \text{Loc\#}, \text{epoch\#}) \\ & \otimes \cdots \otimes H(x_n, n, \text{Type\#}, \text{Loc\#}, \text{epoch\#}).\end{aligned} \tag{4}$$

(ii) *Mode II Verification*. For Mode II verification, TMC could perform the certificate batch verification as well as signature batch verification.

(1) *Certificate Batch Verification*. Given an aggregated certificate $\text{Cert}_{\text{agg}} \leftarrow (i, X_i, \sigma_{\text{agg}})$, the verifier accepts if $e(\prod_{i=1}^{n} \sigma_i, g) = e(\prod_{i=1}^{n} h_i, pk)$ holds.

(2) *Signature Batch Verification*. Given MAC_{agg}, the message set $\text{SSR}_i \mid 1 \leq i \leq n$ and public keys $X_i \mid\leq i \leq n$ for all the vehicles in set $\mathscr{V}$ accept if $e(\text{MAC}_{\text{agg}}, g) = e(a, \prod_{i=1}^{n} X_i) \times e(b, \prod_{i=1}^{n} X_i^{w_i})$.

If the batch verification holds, the aggregator will accept SSRs in list $\mathscr{R}$ as valid SSRs. Then the aggregated SSR in $\mathscr{R}$ will be forward propagated. Meanwhile, the aggregator will put all the SSRs in $\mathscr{R}$ to message list $\mathscr{F}$.

However, the previous proposed solution may face the following two problems. Firstly, it need to carry the original input of each sensing node for future verification. This is because MACs authentication requires the original input. Secondly, the duplicated message should be carefully removed from the aggregation; otherwise many of them will be aggregated for several times. This point is difficult to prevent in the context of VANET, which is a typically dynamic and distributed environment. In the next section, we will introduce a probabilistic data aggregation scheme which could automatically filter duplicate messages.

6. A Probabilistic Data Aggregation Scheme for Vehicular Sensing Networks

In this section, we firstly introduce the concept of FM sketch, which is the foundation of probabilistic data aggregation in vehicular networks. We then propose a secure data aggregation scheme based on our proposed sketch proof technique.

6.1. FM Sketches-Based Data Aggregation in VANETs. A Flajolet-Martin sketch (or "FM sketch") is a data structure for probabilistic counting of distinct elements that has been introduced in [23]. FM sketch represents an approximation of a positive integer by a bit field $s = s_1, \ldots, s_w$ of length w, where $w \geq 1$. The bit field is initialized to zero at all positions. To add an element x to the sketch, it is hashed by a hash function h with geometrically distributed positive integer output, where $P(h(x) = i) = 2^{-i}$. The entry $s_{h(x)}$ is then set to one. After processing all objects, FM finds the first bit of the sketch that is still 0. Let the position of this bit be k; then the number of distinct objects is estimated as $n = 1.29 \times 2^k$.

The variance of n is quite significant [13], and thus, the approximation is not very accurate. To overcome this, instead of using only one sketch, a set of sketches can be used to represent a single value to achieve trade-off between the accuracy and memory. The respective technique is called probabilistic counting with stochastic averaging (PCSA) in [23]. With PCSA, each added element is first mapped to one of the sketches by using an equally distributed hash function, and it is then added there. If m sketches are used, denoted by $S_1, \ldots, S_m$, let $a_1, a_2, \ldots, a_m$ be the positions of the first 0 in the m sketches, respectively; the estimate for the total number of distinct items added is then given by $n = 1.29 \times 2^{k_a}$, where $k_a = (1/m)\sum_{i=1}^{m}(a_i)$.

Sketches can be merged to obtain the total number of distinct elements added to any of them by a simple bitwise OR. Important here is that, by their construction, repeatedly combining the same sketches or adding already present elements again does not change the results, no matter how often or in which order these operations occur. FM sketch summaries are naturally composable: simply OR-ing independently built bitmaps (e.g., over data sets a_1 and a_2) for the same hash function gives precisely the sketch of the union of the underlying sets (i.e., $a_1 \cup a_2$). This makes FM sketches ideally suited for VANET aggregation.

For the purpose of discussion, let us consider a specific application. Assume that we are interested in monitoring the average speed within a certain area. As the first step, we use a sketch for each road segment and approximate the sum of speeds of vehicles within this road segment. For the second step, we will calculate the average speed by dividing the speed sum by the number of vehicles involved. In the following sections, we will discuss how to generate the sketch proof and secure sketch aggregation.

6.2. Sketch Proof Generation. According to the FM sketch definition, given the ID i and speed v_i, a vehicle may add the tuples $(i, 1), \ldots, (i, v_i)$ to the sketch by hashing them and setting the respective bit position $h(i, 1), \ldots, h(i, v_i)$ to 1. The malicious attackers may launch two kinds of attacks towards the FM sketch: inflation attack and deflation attack.

We start from three basic pieces of information that each sensor generates in our protocol. Let $\Lambda^i = \{\ell_1, \ldots, \ell_{v_i}\}$ denote v_i 1-bit positions generated by i. Given that ψ_i is the position of first 0-bit, Λ^i could be represented as the union of two subsets $\Lambda^i_{\psi_i} = \{1, \ldots, \psi_i - 1\}$ and $\overline{\Lambda^i_{\psi_i}} = \{\ell_{\psi_i}, \ldots, \ell_{v_i}\}$, where ℓ_{ψ_i} represents the first 1-bit larger than ψ_i. Thus, each vehicle i generates

(1) $s_i^+ = \{i, \psi_i, \text{Loc\#}, \text{epoch\#}, \text{MAC}_{K_i}(\omega \parallel \text{Loc\#} \parallel \text{epoch\#}) \mid \omega \in \Lambda^i_{\psi_i}\}$, which is called vehicle i's *inflation-free proof*. Here, Loc# and epoch# refer to the road segment number and time slot number, respectively.

(2) $s_i^- = \text{MAC}_{K_i}(\text{Loc\#} \parallel \text{epoch\#})$, which is called vehicle i's *deflation-free proof*. This is basically the authentication code generated by the vehicle on the common information Loc#, epoch#.

(3) $s_i^\times = \{\overline{\Lambda^i_{\psi_i}}, \text{MAC}_{K_i}(\omega \parallel \text{Loc\#} \parallel \text{epoch\#}) \mid \omega \in \overline{\Lambda^i_{\psi_i}}\}$, which is called vehicle i's *supplement security proof*.

In the following, we will introduce these three security proofs one by one.

6.2.1. Inflation-Free Proof. Inflation-free proof is basically the authentication code generated by the vehicles on the 1-bit positions, which are smaller than the position of first 0. To prevent the inflation attacks, it is sufficient to require that each 1-bit, whose position is less than ψ_i, should be authenticated by a single signed value from one of the sensing vehicles that turn it on. We define two extra operations for inflation-free proofs.

(i) *Merging Operation* $\oplus$. Consider two sketches Λ^i and Λ^j (for simplicity of presentation, we assume $\psi_i > \psi_j$). Let ψ_m be the globally maximum value of first 0-bit after sketch merging, which corresponds to the new $\Lambda_{\psi_m} = \{1, \ldots, \psi_m - 1\}$ and $\overline{\Lambda_{\psi_m}} = \Lambda^i \cup \Lambda^j \setminus \Lambda_{\psi_m}$. We define

$$\oplus_{\omega=i,j} s^+_{\psi_\omega} = s^+_{\psi_i} \cup s_i^\times\left(\Lambda_{\psi_m}\right) \cup s_j^\times\left(\Lambda_{\psi_m}\right), \tag{5}$$

where $s_i^\times(\Lambda_{\psi_m})$ is the operation that picks up all the supplement security proof whose positions are less than ψ_m. In other words, to generate inflation-free proof for the merged sketches, the aggregator could first pick up the inflation-free proof $s^+_{\psi_i}$ of the sketch with a higher 0-bit position ψ_i. For the remaining 1-bit positions $\psi_i, \ldots, \psi_m - 1$, the aggregator could pick up the inflation-free proofs either from $s_i^\times$ or $s_j^\times$. Note that, if a 1-bit is authenticated by multiple MACs generated by multiple vehicles, aggregators could choose inflation-free proof of vehicles with a lower ID.

(ii) *Aggregation Operation* $\otimes$. The MACs of s_i^+ could be further aggregated. For example, if MAC is generated by symmetric key-based hash function (e.g., MD5 or SHA-1), then ⊗ can be simple XOR; if MAC is signatures, ⊗ could be achieved by using aggregate signature technique such as [19].

With merging operation and aggregation operation, size of inflation-free proof could be minimized to $|\text{ID}| * N_{1-\text{bit}} + |\text{MAC}|$, where |ID| and |MAC| refer to the size of vehicle ID and MAC, respectively, and $N_{1-\text{bit}}$ denotes the number of 1-bits.

6.2.2. Deflation-Free Proof. Deflation attack is defined as that the malicious aggregators may try to turn 1-bits into 0-bits, removing the corresponding MACs from the security proofs. To prevent deflation attack, SAS adopts the hash-chain-based MAX protocol, which is introduced in [16]. The basic idea is to construct one-way chains whose seeds are all the s_i^-. Specifically, given the one-way function $F()$, vehicle node i reports to the aggregator $F^{\psi_0}(s_i^-)$. In a case of multiple sketch aggregation, let ψ_m be the maximum positions observed by the aggregator. The aggregator can obtain $F^{\psi_m}(s_i^-)$ by performing hash operations on $F^{\psi_0}(s_i^-)$ by $\psi_m - \psi_0$ times. After obtaining all the $F^{\psi_m}(s_i^-)$, a new operation is introduced in [16] to reduce the transmission cost, which is shown as follows.

(i) *Hash Chain Folding Operation* ⊙. The aggregator could use the folding function ⊙ to fold all the hash chains into a single one $\odot F^{\psi_m}(s_i^-)$. Obviously, due to adoption of one-way function, it is impossible for the attackers to generate a new security proof for $\psi_i < \psi_m$, which prevents the deflation attack.

Note that one-way chains should be rolled forward even after they have been folded together with an operation like ⊙. Therefore, it requires the one-way function to achieve homomorphic property in that $F(x_1 \odot x_2) = F(x_1) \odot F(x_2)$. There is a wide range of cryptographic tools such as RSA encryption that could support such kind of homomorphic property. In this case, ⊙ could be defined as modular multiplication.

The size of deflation-free proof is a constant number $|F()|$, which represents the size of one-way function output. If choosing RSA as the cryptographic tool, $|F()| = 1024$.

6.2.3. Supplement Security Proof. Supplement security proof enables the aggregator to derive the new inflation-free proof when ψ_0 changes because of the merge of sketches. Therefore, SAS records all 1-bits whose positions are larger than ψ_m and their corresponding MACs as the supplement security proof. Since they are not continuous, supplement security proof cannot be aggregated. Further, we denote $s_i^{\times}(\overline{\Lambda_{\psi_m}})$ as the set of all the supplement security proofs whose positions are not less than ψ_m.

6.3. Sketch Proof Aggregation. As shown in Figure 2, multiple sketches could be aggregated during their propagation process, and sketch proofs could be aggregated along with sketches merging. Without loss of generality, we discuss aggregation algorithm only for two sketch proofs, and more than two sketch aggregations can be aggregated by applying it for multiple times.

Consider two sketches Λ^i and Λ^j and their corresponding sketch proofs $s_i^+, s_i^-, s_i^{\times}$ and $s_j^+, s_j^-, s_j^{\times}$. Let ψ_m be the globally maximum value of first 0-bit after sketch merging. The sketch proofs could be aggregated by performing the following steps:

(i) inflation-free proof aggregation: $\otimes(\oplus_{\omega=i,j} s_{\psi_w}^+)$;

(ii) deflation-free proof aggregation: $\odot_{\omega=i,j} F^{\psi_m}(s_\omega^-)$;

(iii) supplement security proof updating:

$$s_i^{\times}\left(\overline{\Lambda_{\psi_m}}\right) \cup s_j^{\times}\left(\overline{\Lambda_{\psi_m}}\right). \quad (6)$$

Note that such a sketch proof aggregation process could be performed fully distributed, which means it naturally supports hierarchical aggregation, while it does not require any aggregation architecture.

6.4. Sketch Proof Verification and Average Calculation. After the aggregation results and the security proof arrive at the TMC, TMC should verify the correctness of the inflation-free proof and deflation-free proof. To check the validity of inflation-free proof, TMC should perform the following operations in different MAC modes.

(i) *Symmetric Key Mode*. In this mode, TMC should recalculate the MAC of each 1-bit and then aggregate them into a single one. After that, TMC should check if the obtained result is equal to the received one.

(ii) *Signature Mode*. In this mode, TMC could batch verify the aggregated signatures by performing batch verification technique [19].

To verify the correctness of deflation-free proof, it needs to compute all individual s_ω^- and fold them together to create the $\odot_{\omega=1,2,\ldots} F^{\psi_m}(s_\omega^-)$. The answer is accepted if and only if the calculated result is equal to the received one. Finally, by obtaining the ψ_m, the average speed could be computed as follows:

$$\text{speed}_{\text{average}} = 1.29 \times \frac{2^{\psi_m}}{N_{\text{ID}}}, \quad (7)$$

where N_{ID} refers to the number of vehicles involved. Similar to the original FM sketch, the accuracy of this average speed estimation could be further improved by introducing multiple sketches.

6.5. Further Discussion. In this subsection, we give an extended discussion on some issues closely related to the proposed SAS protocol.

6.5.1. Symmetric Key versus Asymmetric Key. As we have mentioned in Section 3, MAC in this study represents two modes: symmetric key-based mode and asymmetric key- (or signature-) based mode. Generally speaking, different MAC modes have different advantages as well as disadvantages.

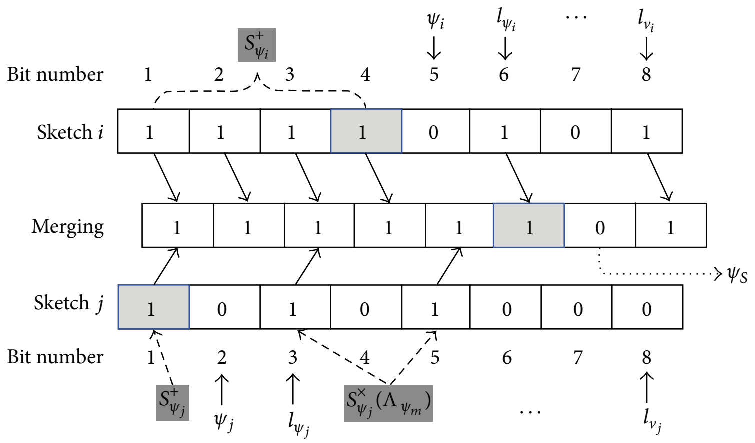


Figure 2: Sketch generation and sketch proof.

From the performance point of view, symmetric key-based MAC has the advantage on asymmetric key-based approach in that it has shorter size and will not introduce the computational expensive operations. Symmetric key-based MAC is expected to play an important role in the vehicular sensing applications where sensing information is directly sent to the TMC since they could be processed faster than signature-based approach and also introduce less transmission overhead. However, on-path vehicles cannot verify an MAC's authenticity since only TMC shared the key with MAC generator. On the other hand, signature-based approach could provide many extra features such as nonrepudiation and public authentication. In the context of vehicular sensing networks, it means the aggregated information could be verified by any on-path vehicles, which allows the drivers to have fast access to the authenticated traffic information instead of waiting for the response of the RSUs.

6.5.2. Size of Sketch Proofs. There are three kinds of sketch proofs for SAS. The first two sketch proofs including inflation-free proof and deflation-free proof could be aggregated and thus introduce a minimized transmission overhead. The third sketch proof, supplement security proof, does not support proof aggregation since they will be merged with inflation-free proof in the future. This means supplement security proof may incur a higher transmission overhead. However, we argue that size of supplement security proof is still acceptable in that, during the aggregation process, size of supplement security proof will decrease along with the increase of first 0-bit position ψ_m. In the performance evaluation part, we will give a more detailed discussion on the size of sketch proofs.

7. Performance Evaluations

In this section, we evaluate the performance of the proposed SAS in terms of the resultant communication cost and approximate accuracy. To demonstrate the superiority of SAS, we also compare SAS with nonaggregation transmission approach. In this part, we consider SHA-1 as the building blocks of MAC. Note that asymmetric key-based MAC mode will have a similar communication cost if we choose short aggregate signature as the building blocks.

Table 1: The size of each component of SAS (bytes).

	No SAS	SAS
T&L	$8 \times n$	8
ID	$8 \times n$	$8 \times n$
Data v	$8 \times n$	0
Sketch$_i$	0	$8 \times \log_2(v_{\max} \times n)$
Sketch proofs	$8 \times n$	$8 \times \log_2(v_{\max} \times n) + 136$
Total size	$32 \times n$	$8 \times n + 16 \times \log_2(v_{\max} \times n) + 144$

7.1. Transmission Overhead. One of the major advantages of SAS is the reduction of its transmission cost. The communication cost is determined by the size of aggregated security proof including inflation-free proof, deflation-free proof, and supplement security proof. Note that, since MAC in this study represents two modes: symmetric key-based mode and asymmetric key- (or signature-) based mode, here we only discuss the symmetric key-based MAC due to page limitation. As a typical example, we choose the 64-bits SHA-1 as the basic MAC technique and RSA-1028 as the basic one-way function tool. Table 1 summarizes the size of different components as well as the overall transmission overhead for nonaggregation transmission and SAS transmission. Here, we consider the worst case of our aggregation in that the size of supplement security proofs is bounded by $\log_2(v_{\max} \times n)$ [13], where $v_{\max}$ is the maximum speed for this road segment while n is maximum number of vehicles in this area. However, it is important to point out that, in practice, the size for supplement security proof should be much less than this bound since it will decrease along with the aggregation.

By choosing different number of sketches, we obtain the different communication cost of SAS under different vehicle numbers as well as different sketch numbers, which has been shown in Figure 3. It is observed that the probabilistic aggregation does not show its advantage when the

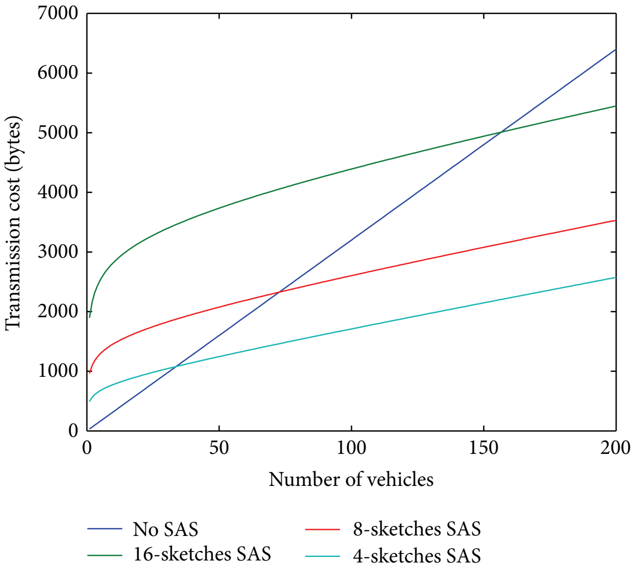


Figure 3: Transmission overhead of various secure FM sketches.

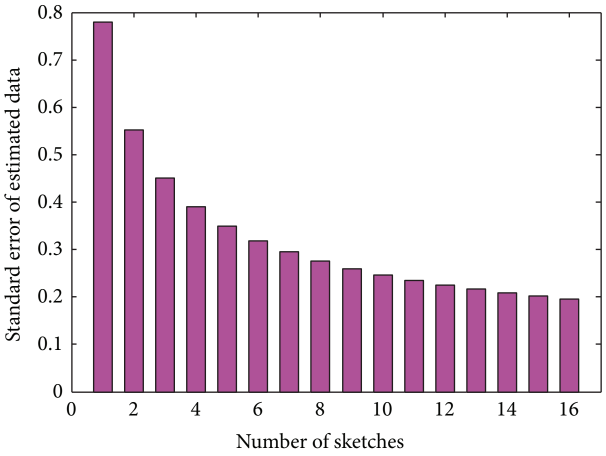


Figure 4: Standard error of SAS secure sketch.

number of vehicles is small. However, when the number of vehicles grows, the proposed SAS aggregation scheme could dramatically reduce the communication cost when the sketch number is small. It is also observed that the number of sketches plays an important role for the overall system performance in that a small sketch number such as 4 makes the proposed SAS have a better performance while, when the sketches number is large such as 16, the advantage is not so obvious. Therefore, if an acceptable accuracy is guaranteed, the number of sketches should be as small as possible to achieve a better performance. In the next section, we will discuss the tradeoff of accuracy and the number of sketches.

7.2. Tradeoff of the Accuracy and Number of Sketches. According to [13], PCSA yields a standard error of approximately $0.78/\sqrt{m}$. By choosing different sketch numbers, we can obtain the corresponding standard error, which has been plotted in Figure 4. It is observed that the standard error decreases dramatically along with the increase of number of sketches in the beginning while it stays relatively stable after a specific threshold (e.g., 4 in Figure 4). However, as we pointed out in the previous section, in the vehicular sensing networks, a small number of sketches (e.g., 4) guarantee an acceptable standard error (e.g., 0.39). This further demonstrates the effectiveness of the proposed SAS.

8. Conclusion and Future Work

Vehicular sensing networks have been envisioned to play an important role for future traffic monitoring applications. In this study, we propose a secure and efficient aggregation method based on FM sketch and security proofs techniques. The extensive performance evaluations have demonstrated the efficiency and effectiveness of the proposed scheme. Our future work includes implementing SAS in a specific application scenario and evaluating its performance with more realistic simulations or even experiments.

Acknowledgments

The correspondence author of this paper is H. Zhu. This research is supported by the National Natural Science Foundation of China (Grant no. 61003218, 70971086, 61272444, 61161140320, and 61033014), the Doctoral Fund of Ministry of Education of China (Grant no. 20100073120065), the JSPS A3 Foresight Program, and the NEC C&C Foundation.

References

[1] H. Zhu, X. Lin, M. Shi, P. H. Ho, and X. Shen, "PPAB: a privacy-preserving authentication and billing architecture for metropolitan area sharing networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 5, pp. 2529–2543, 2009.

[2] H. Zhu, S. Du, Z. Gao, M. Dong, and Z. Cao, "A probabilistic misbehavior detection scheme towards efficient trust establishment in delay-tolerant networks," *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[3] M. Fontaine, "Traffic monitoring," in *Vehicular Networks from Theory to Practice*, CRC Press, 2009.

[4] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen, "SMART: a secure multilayer credit-based incentive scheme for delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628–4639, 2009.

[5] H. Zhu, R. Lu, X. Shen, and X. Lin, "Security in service-oriented vehicular networks," *IEEE Wireless Communications*, vol. 16, no. 4, pp. 16–22, 2009.

[6] H. Zhu, X. Lin, R. Lu, P. H. Ho, and X. S. Shen, "SLAB: a secure localized authentication and billing scheme for wireless mesh networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 10, pp. 3858–3868, 2008.

[7] A. Liu, Z. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multipath routing for WSNs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3255–3265, 2012.

[8] A. Liu, X. Jin, G. Cui, and Z. Chen, "Deployment guidelines for achieving maximal lifetime and avoiding energy holes in sensor network," *Information Sciences*, vol. 230, pp. 197–226, 2013.

[9] A. Liu, J. Ren, X. Li, Z. Chen, and X. Shen, "Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks," *Computer Networks*, vol. 56, no. 7, pp. 1951–1967, 2012.

[10] A. Liu, P. Zhang, and Z. Chen, "Theoretical analysis of the lifetime and energy hole in cluster based wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 71, no. 10, pp. 1327–1355, 2011.

[11] M. H. Arbabi and M. Weigle, "Using vehicular networks to collect common traffic data," in *Proceedings of the 6th ACM International Workshop on Vehicular Ad Hoc Networks (VANET '09)*, pp. 117–118, Beijing, China, 2009.

[12] S. Dietzel, B. Bako, E. Schoch, and F. Kargl, "A fuzzy logic based approach for structure-free aggregation in vehicular ad-hoc networks," in *Proceedings of the 6th ACM International Workshop on Vehicular Ad Hoc Networks (VANET '09)*, pp. 79–88, Beijing, China, 2009.

[13] C. Lochert, B. Scheuermann, and M. Mauve, "Probabilistic aggregation for data dissemination in VANETs," in *Proceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks (VANET '07)*, pp. 1–8, September 2007.

[14] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: traffic data dissemination using car-to-car communication," *Mobile Computing and Communications Review*, vol. 8, no. 3, pp. 6–19, 2004.

[15] M. Caliskan, D. Graupner, and M. Mauve, "Decentralized discovery of free parking places," in *Proceedings of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET '06)*, pp. 30–39, ACM, New York, NY, USA, September 2006.

[16] S. Nath, H. Yu, and H. Chan, "Secure outsourced aggregation via one-way chains," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '09)*, pp. 31–44, Providence, RI, USA, 2009.

[17] H. Zhu, X. Lin, R. Lu, P. H. Ho, and X. Shen, "AEMA: an aggregated emergency message authentication scheme for enhancing the security of vehicular ad hoc networks," in *Proceedings of IEEE International Conference on Communications (ICC '08)*, pp. 1436–1440, Beijing, China, May 2008.

[18] M. Raya, A. Aziz, and J. P. Hubaux, "Efficient secure aggregation in VANETs," in *Proceedings of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET '06)*, pp. 67–75, September 2006.

[19] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weilpairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[20] J. Camenisch, S. Hohenberger, and M. Pedersen, "Batch verification of short signatures," in *Advances in Cryptology (EUROCRYPT '07)*, vol. 4515 of *Lecture Notes in Computer Science*, pp. 246–263, Springer, New York, NY, USA, 2007.

[21] T. D. C. Little and A. Agarwal, "An information propagation scheme for VANETs," in *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 155–160, September 2005.

[22] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: a mobility-centric data dissemination algorithm for vehicular networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET '04)*, pp. 47–56, October 2004.

[23] P. Flajolet and G. N. Martin, "Probabilistic counting algorithms for data base applications," *Journal of Computer and System Sciences*, vol. 31, no. 2, pp. 182–209, 1985.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 764064, 9 pages
http://dx.doi.org/10.1155/2013/764064

Research Article

An Improved Trust Model Based on Interactive Ant Algorithms and Its Applications in Wireless Sensor Networks

Yun Pan, Yafang Yu, and Li Yan

School of Computer Science, Communication University of China, Beijing 100024, China

Correspondence should be addressed to Yun Pan; pany@cuc.edu.cn

Received 4 January 2013; Revised 29 March 2013; Accepted 3 May 2013

Academic Editor: Motoki Sakai

Marmol et al.'s ant algorithm based trust model is improved from several aspects: new interactive ant algorithm, new node types, more reasonable meta-assumption on node behaviors, new trust evaluation function, new penalty mechanism, and so on. Simulations on identifying malicious nodes and electing cluster head show that the proposal is effective and can observably reduce the packet drop ratios.

1. Introduction

Wireless sensors are small and cheap devices powered by low-energy batteries, equipped with radio transceivers, and responsible for monitoring physical or environmental conditions, such as temperature, humidity, sound, pressure, motion, and anything we are interested in. They are featured with resource (e.g., power, storage, and computation capacity) constraints and low transmission rates [1]. Wireless sensor networks (WSNs) are networks based on such wireless sensors cooperation. They can be used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment monitoring, healthcare applications, home automation, and traffic control.

However, the wireless sensor nodes are usually deployed in open environment where an adversary may easily capture sensor nodes and subsequently use these nodes to attack the whole network [1]. Therefore, it is desirable to identify the compromised nodes in time and then kick them out so as to avoid the whole network to be controlled by the adversary [2]. Traditional security system is usually based on cryptography, which requires complexity encryption and decryption operations. But some cryptographic countermeasures are not efficient and smart enough to be deployed in wireless sensor networks where sensors have limited communication bandwidth, memory, and computing power [3]. Although research on lightweight cryptography makes great progress on dealing with this problem, many currently available cryptographic components are far from lightweight. For example, the reported most lightweight implementation of elliptic curve cryptography (ECC) requires about 10,000 GEs (i.e., equivalent gates), but in general there is no more than 6000 GEs that are left for deploying security in typical sensors.

Therefore, it is interesting to probe effective non-cryptographic mechanism for identifying corrupted nodes, and then take the corresponding countermeasures to elevate the losses. Trust management technique is regarded a good complementary toward the system security protections based on cryptographic mechanism [3]. Since Marsh introduced the concept of computable trust model, many researchers proposed various trust models for different scenarios. Recently, a lot of trust models for WSN-oriented applications were proposed, such as for securing routing, data aggregation [4], cluster head selection [5], and synthesized trust management systems [6]. All these trust models are featured as classical computational patterns. In 2008, Mármol and Pérez proposed a novel trust model—BTRM-WSN that employs a smart-warm intelligent optimization algorithm—ant algorithm—into trust management model. Although at present any ant algorithm can merely be implemented by adopting classical computational pattern, its typical bionic computational pattern enables it powerful capability for global optimization and this in turn introduces new promising properties on trust management. Although Mármol and Pérez [7] attested the advantages of BTRM-WSN model by using amount of simulations, this model suffers from too many specializations and constraints, resulting in a very narrow scope on its application. Considering that interactive ant algorithms are more effective than noninteractive ones, our main motivation is to, based on Marmol's work, improve the efficiency with interactive multiple ant colony algorithm and extend the suitability of BTRM-WSN model by proposing new improvements from node types, node functionalities and trust value increasing, penalty function, and so forth.

The rest contents are organized as follows. In Section 2, we give an introduction on the new trust model based the BTRM-WSN model and our improvements; in Section 3, we conduct detailed simulations on our trust model; in Section 4, as a typical application of the proposed model, we, based on interactive ant colony algorithm, present a trust cluster head election framework for WSN environments. Finally, the concluding remarks are given in Section 5.

2. Trust Model

In [7], the authors presented a trust model for WSNs, called BTRM-WSN, based on ant colony systems aiming to help a node requesting a certain service to the network and find the most trustworthy route leading to a node providing the right requested service. Experiments and results demonstrated the accuracy and robustness of this model. Based on the original BTRM-WSN model, we introduce the following improvements to enhance its efficiency and scope.

2.1. Interactive Multiple Ant Colony Algorithm (IMACA). Interacted Multiple Ant Colonies Optimization method just like Ant Colony System (ACS) is a bioinspired algorithm. In IMACA there are also two levels of interaction. One is the colony level and the other one is the population level [8].

The activities of a single colony in IMACA method are based on ACS. Each colony has its own pheromone that is used as an interaction between the ants of the same colony. The interaction between ant colonies using pheromone can be organized in different terms [8]. The IMACA algorithm is described as follows. M colonies of m ants each are working together to solve some combinatorial problem. Let ant k which belongs to the colony v be at node r at a certain moment. The probability of moving towards node $s \in N_i^{\mathrm{kv}}$, where N_i^{kv} is the set of remaining neighbors not visited yet by ant kth ant of colony v, is computed as

$$p_{\mathrm{k}}^{\mathrm{v}}(\mathrm{r}, \mathrm{s}) = \begin{cases} \arg\max\limits_{\mathrm{u} \in N_{\mathrm{r}}^{\mathrm{kv}}} \left\{ f(P_{\mathrm{ru}}) H_{\mathrm{ru}}^{\beta} \right\} & \text{if } q \le q_0 \\ S & \text{otherwise,} \end{cases} \qquad (1)$$

where $f(P_{\mathrm{rs}})$ is the evaluation function of pheromone on the edge (r, s), H_{rs} is the problem dependent heuristic, β is a value used to determine the relative importance of pheromone versus heuristic, $q_0 \in [0, 1]$ is a constant, and $q \sim [0, 1]$ is a random number within the interval $[0, 1]$, while S is a random variable selected according to the following probabilistic formula:

$$S = \begin{cases} \dfrac{f(P_{\mathrm{rs}}) H_{\mathrm{rs}}^{\beta}}{\sum_{\mathrm{u} \in N_{\mathrm{r}}^{\mathrm{kv}}} f(P_{\mathrm{ru}}) H_{\mathrm{ru}}^{\beta}} & \text{if } \mathrm{s} \in N_{\mathrm{r}}^{\mathrm{kv}} \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

If $q \le q_0$, the most promising node is selected as the next step of ant k using expression (1); otherwise, that node is chosen using expression (2).

The pheromone evaluation function of IMACA evaluates the pheromone on an edge as a composition between the pheromone values of the ant own colony and the value of the pheromone evaluation function based on some pheromone evaluation rate [8]. An ant builds γ (γ is the pheromone evaluation rate between $[0, 1]$) of its decision based on its own colony's experience and the other based on others. The pheromone evaluation function is computed as

$$f(P_{\mathrm{rs}}) = \gamma P_{\mathrm{rs}}^{S} + (1 - \gamma) \frac{\sum_{\mathrm{v}=1}^{\mathrm{M}} P_{\mathrm{rs}}^{\mathrm{v}}}{\mathrm{M}}, \qquad (3)$$

where $P_{\mathrm{rs}}^{\mathrm{v}}$ is the pheromone of colony v on the edge (r, s).

Just like ACS, there are two kinds of pheromone updating: a local and a global one. Local pheromone update is then applied by each ant on the visited edges. The local pheromone update is defined as

$$P_{\mathrm{rs}}^{\mathrm{v}} = (1 - \varphi) P_{\mathrm{rs}}^{\mathrm{v}} + \varphi p_0, \qquad (4)$$

where φ is a pheromone evaporation parameter and p_0 is the initial pheromone value.

Global pheromone updating includes that the best ant of each colony deposits an amount of pheromone on its own path. The ant can find the best solution according to the follpwing rule:

$$P_{\mathrm{rs}}^{\mathrm{v}} = (1 - \rho) P_{\mathrm{rs}}^{\mathrm{v}} + \rho \Delta P_{\mathrm{rs}}^{\mathrm{v.bs}} \qquad (5)$$

$\rho \in [0, 1]$ is the trail evaporation parameter and $(1 - \rho)$ represents the pheromone persistence. $\Delta P_{\mathrm{rs}}^{\mathrm{v.bs}}$ is the pheromone quantity added to the connection (r, s) belonging to the best solution of vth colony $L^{\mathrm{v.bs}}$ and is computed as [8]

$$\Delta P_{\mathrm{rs}}^{\mathrm{v.bs}} = \begin{cases} \dfrac{1}{L^{\mathrm{v.bs}}} & \text{if } (\mathrm{r}, \mathrm{s}) \text{ belongs to the best tour of colony v} \\ 0 & \text{otherwise.} \end{cases} \qquad (6)$$

2.2. New Trust Model Using IMACA. By adopting new interactive ant algorithm IMACA described above, we propose a new trust model that can be viewed as an improvement on Mármol and Pérez original model BTRM_WSN [7].

2.2.1. Node Types. In the original mode, nodes are divided into two disjoint sets: the set of client nodes and the set of server nodes. This separation exerts many constraints for

WSNs by requiring that server nodes cannot request services and client nodes cannot provide services. However, in a distributed ad hoc network, each node could be server and client. In our improved model, the node types are aggregated such that a node can simultaneously request services from others and provide services for others. Apparently, this model is even more suitable for actual WSNs.

2.2.2. Node Behavior. In the original model, the behavior of malicious nodes is assumed to provide malicious services but does not specify what kind of malicious services is, and does not consider the infectious factors of packet loss rate of a node. In addition, in the original model, all client nodes are assumed to be trusted and without fault behavior. This makes the original model far from the real WSN environments. Therefore, according to node behaviors described in [3, 9, 10], a malicious is specified a very high packet loss rate (between 0% and 40%), while the packet loss rate of a trust node is specified between 0% and 10%.

2.2.3. Diagram of the New Trust Model. Based on the improvement of node types and node behavior, before a node requests to another node, it would perform the following steps.

(1) Let many ants of multiple ant colonies search path. These ants select the next step according to expressions (1) and (2).

(2) When ants select the next hop node, update pheromone value of the edge to the selected node. Every time an ant moves from one node to another, the pheromone local updating is carried out through the following expression:

$$P^{\mathrm{v}}_{s_1 s_2} = \left(1-\varphi\right)\cdot P^{\mathrm{v}}_{s_1 s_2} + \varphi\cdot\Omega, \tag{7}$$

where

$$\Omega = \left(1 + \left(1-\varphi\right) * \left(1 - P^{\mathrm{v}}_{s_1 s_2} H_{s_1 s_2}\right)\right)\cdot P^{\mathrm{v}}_{s_1 s_2}. \tag{8}$$

After all ants find the best path, global updating is applied on those edges belonging to this path by using the following expression:

$$P^{\mathrm{v}}_{\mathrm{rs}} = \left(1-\varphi\right)\cdot P^{\mathrm{v}}_{\mathrm{rs}} + \varphi * \left(1 + P^{\mathrm{v}}_{\mathrm{rs}} H_{\mathrm{rs}} \Delta P^{\mathrm{v.bs}}_{\mathrm{rs}}\right)\cdot P^{\mathrm{v}}_{\mathrm{rs}}. \tag{9}$$

At last, merge all the pheromone of all colonies using the following expression:

$$P_{\mathrm{rs}} = \frac{\sum_{\mathrm{v}=1}^{\mathrm{M}} P^{\mathrm{v}}_{\mathrm{rs}}}{\mathrm{M}}. \tag{10}$$

(3) When ants find the node to meet the requirements, return to the initial node in the same way. And record the path information.

(4) Each time a launched ant returns to its node carrying a path or solution. Among all possible paths, that node would like to choose a path that has the best quality. The path quality computation can be done in the following way:

$$Q\left(S_{\mathrm{k}}\right) = \frac{\overline{P_{\mathrm{k}}}}{\sqrt{\mathrm{Length}\left(S_{\mathrm{k}}\right)}}\cdot A_{\mathrm{k}}\%, \tag{11}$$

where $\overline{P_{\mathrm{k}}}$ is the average pheromone of the path found by ant k and $A_{\mathrm{k}}\%$ represents the percentage of ants that have selected the same solution as ant k.

After the node selects the best path, it requests service by the path. If the provided service cannot meet the requirements, the neighbor of the service node reduces the pheromone values with

$$P_{\mathrm{rs}} = \left(P_{\mathrm{rs}} - \varphi\right)\cdot S_{\mathrm{at}}, \tag{12}$$

where $S_{\mathrm{at}} \in [0,1]$ means satisfaction that the node is about the service.

2.2.4. Trust Value Evaluation. In addition, there is no original trust value in BTRM-WSN. Every node maintains a set of pheromone trace s for all neighbors and these pheromone traces will determine the probability of ants choosing a certain route or another. Therefore, the pheromone trace can be treated as the amount of trust. However, in order to distinguish between pheromone and trust value, we defined T_{i} as the trust value of node i, where $1 \le \mathrm{i} \le \mathrm{k}$. And trust value can be computed based on pheromone trace as follows:

$$T = \frac{\sinh\left(\left(6.2+\alpha\right)p - 3.1 - \left(\alpha/2\right)\right)}{\sinh\left(3.1 + \left(\alpha/2\right)\right)}, \tag{13}$$

where $p \in [0,1]$ is the pheromone value of the edge connected to a node, while α means the degree of strictness.

In our model, each node maintains a trust table in which it records the trust value of its neighbors, as is shown in Figure 1. Node i has 3 neighbor nodes m, n, and o. Based on the pheromone trace of each edge, node i's trust table is shown on the right side.

2.2.5. Punish and Reward. After using trust value, the punishment mechanism is also improved. In the original model, the punishments and rewards are designed in order to find a trusted service node, which involves the whole path. For example, the client node C wants to find the optimal path to the service node S: C → A → B → D → S. According to the service provided by S, C determines that the service does not meet its requirements so that this path would be punished by reducing the value of all the edges of the pheromone. If adopt our adding trust value (see Section 2.2.4), the trust value is a single increasing function of the value of the pheromone, and then reducing the pheromone value means to reduce the trust value of a node. If only S node is malicious node and other nodes are trusted node on this path, then the trust node's trust value will be reduced due to the impact of malicious nodes, and such punishment is clearly unreasonable.

Therefore, the entire punishment mechanism will be modified to only reduce the trust value of the malicious

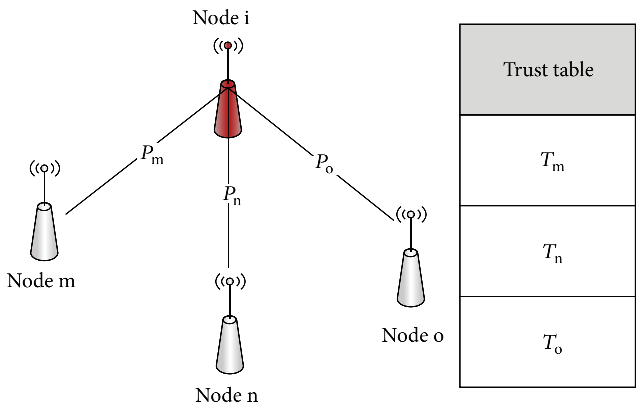


Figure 1: Trust records of node i.

node without an impact on other nodes. In this way, we can guarantee the correctness of the trust value. For ease of description we modify the punishment, when we will use an example to illustrate. Suppose that node C sends the packet to node S, and the improved model throughout the process is as follows.

(1) C sets n ants carrying the control packet to search path. These ants select the next hop node in accordance with the state transition probability formula.

(2) When ants choose the next hop node N, detecting whether the node forward the control packet. If so, the pheromone values leading to the edge of the node will be updated according to pheromone update formula; otherwise, the ants backtrack a hope to node N-1; during the return the node is punished by reducing r edge pheromone value using formula (12). And continue searching the neighbor nodes of the node N-1, until it reaches the destination node S. If the ants have been returned to the node C, all the neighbors of node C which have also been probing finished, and then the ants stop detection.

(3) When the ants find the path leading to the destination node S, backtrack to the node C and record the entire path information (including node information as well as the edge of the pheromone value).

(4) Node C calculates s the quality of the road according to the return path of the ants brought information, and to select the best quality path from all the ants chose the path s.

(5) Node C sends packets to the node S through the optimal path.

(6) After node S received successfully packets, this path will be rewarded that the pheromone values of all edge s of the path will be increased.

2.2.6. Packet Drop Ratio. The packet drop ratio of nodes is introduced in order to make this model more realistic; while in BTRM-WSN, this is never considered as one of the influencing factors. Now the drop ratio of good nodes is 0.001, and bad ones is 0.4. [5, 11].

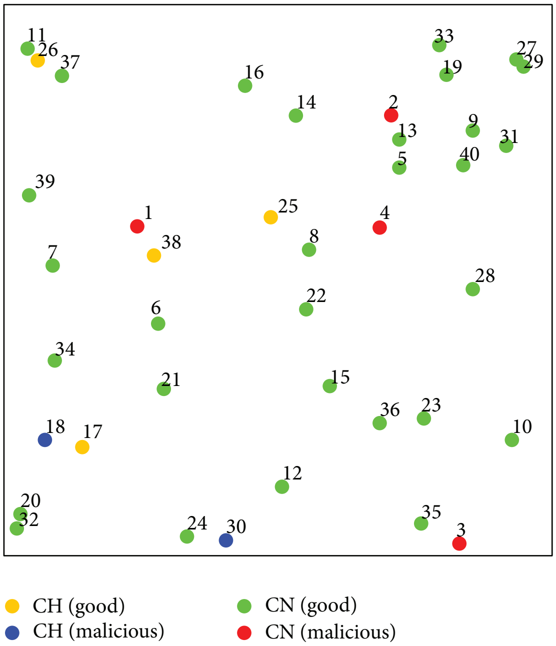


Figure 2: Randomly scattered of nodes.

3. Simulations

In Section 2, we added trust value in the model and set the packet loss rate for nodes. In this section we refer to the experimental method in [6] to evaluate the performance of our improved model from two aspects: the first is to identify malicious nodes and the second is to reduce network packet loss rate. In our model, there is no difference between client nodes and server nodes. So we compare our trust model to Sun et al.'s model [6] instead of BTRM_WSN. The experiment simulation tool is TRMSim-WSN that was developed by Mármol and Pérez for the trust model specifically for wireless sensor network simulation [9].

3.1. Identifying Malicious Nodes. 40 nodes are randomly scattered in the area of 100 m ∗ 100 m (see Figure 2). Each over a certain time, the node randomly selects a node as a destination node to send data packets, respectively, in accordance with this improved model and Sun model.

3.2. Capability of Dynamically Identifying Malicious Nodes. In this experiment, we initially set nodes 1, 2, 3, 4 as the malicious nodes and assume that other 4 nodes will become malicious over every given time segments. That is, after 1 segment, the nodes 5, 6, 7, 8 become malicious; after 2 segments, nodes 9, 10, 11, 12 become malicious. Figure 3 shows the average pheromone value of the nodes in the network run this improved model obtained in different time periods the running Sun et al.'s model obtained in different time segments. The average probability value is shown in Figure 4.

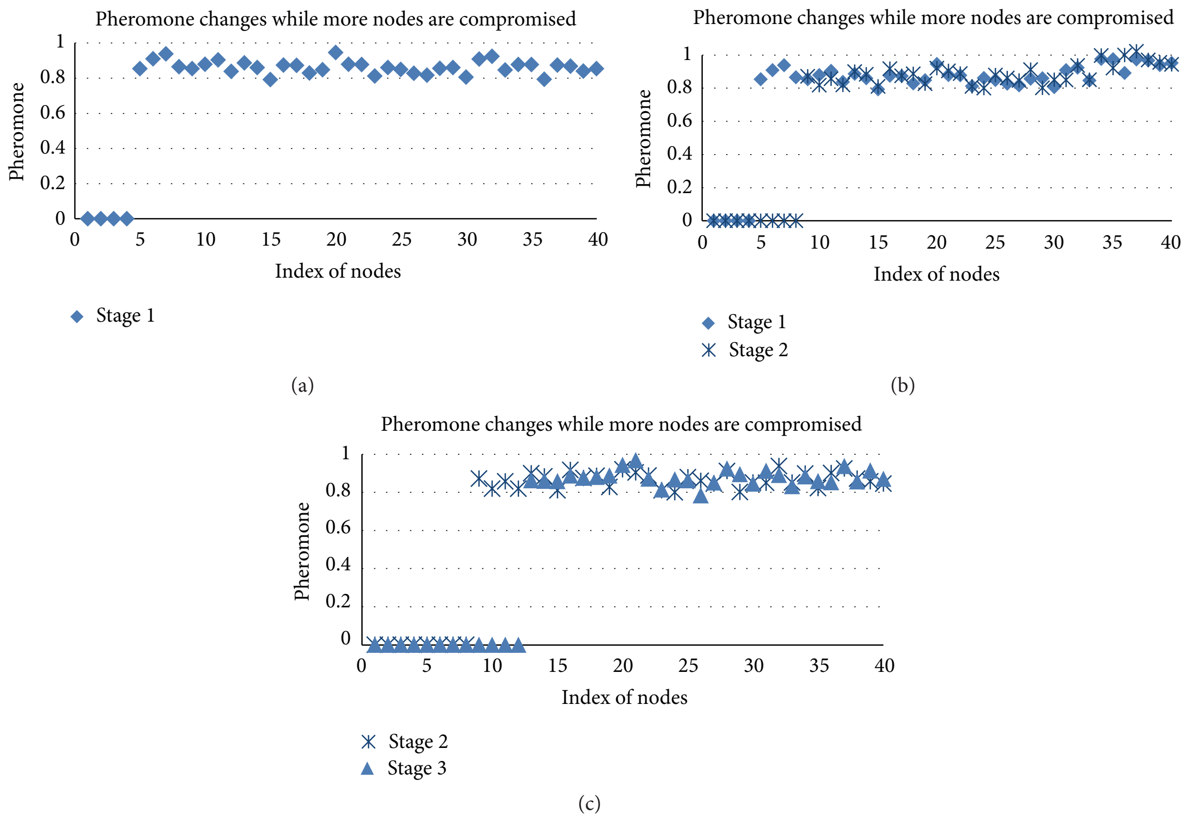


Figure 3: Capability of the improved model for detecting malicious nodes.

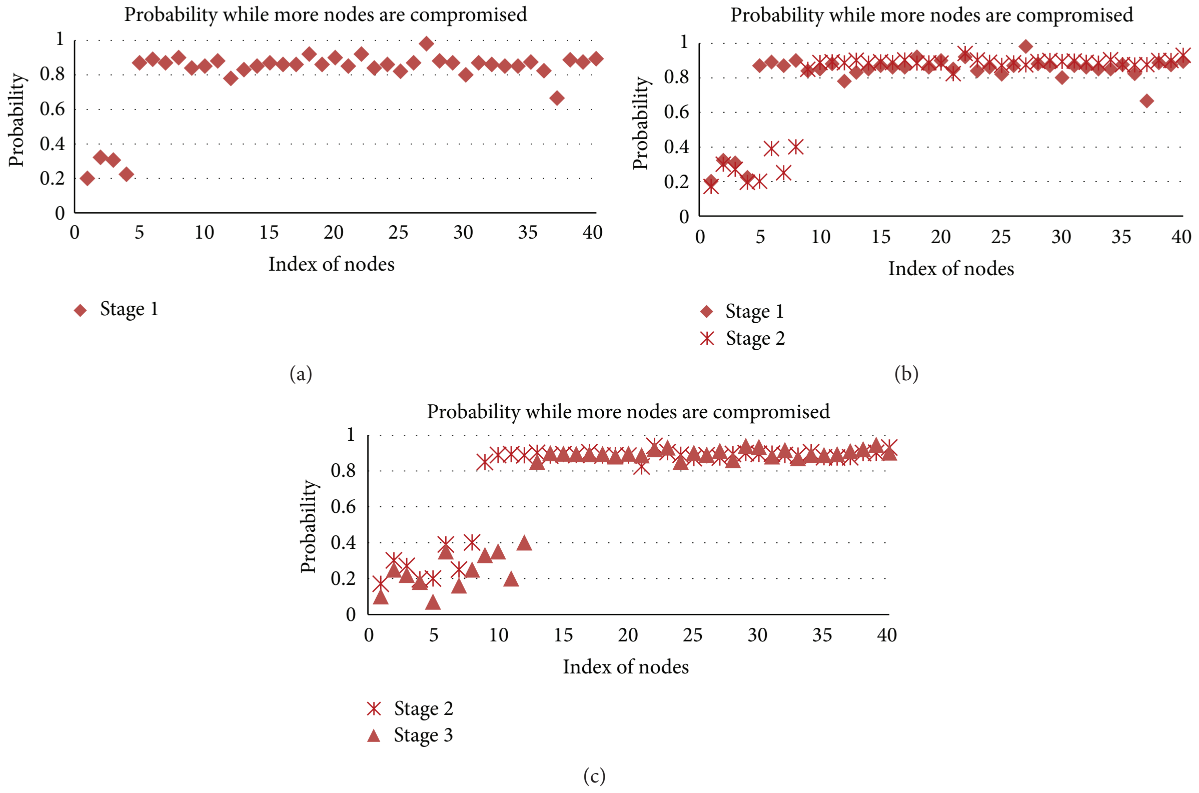


Figure 4: Capability of Sun's model for detecting malicious nodes.

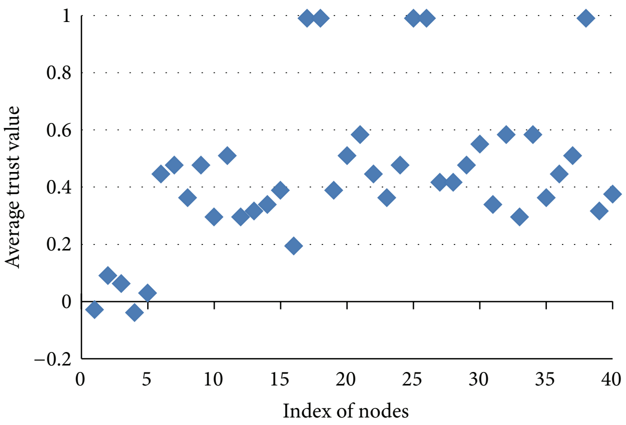


Figure 5: Average trust value of nodes.

Comparing these two graphs, we can find that their pheromone value or probability value will have a very significant change whenever a new node becomes malicious. In particular, the node's pheromone value change range in our improved model is much larger than that of in Sun et al.'s model. The reason is that there is no punishment mechanism in the Sun's model, the trust value of the node only positive feedback, while in our improved model, the node's trust value not only positive feedback but also negative feedback, leading to enhancement on the capability of identifying malicious nodes. This also means that even if a malicious node in the network was initially regarded as trusted, but as long as it has a malicious behavior, it will be identified.

3.3. Node's Trust Table. In this experiment, we set the nodes 1, 2, 3, 4, 5 as malicious nodes. Figure 5 shows the average trust value of all nodes, and wherein, x-axis represents a node in the network ID, and the y-axis indicates the average trust value that calculates according to their neighbor nodes keep.

From Figure 5, we can see that the neighbor nodes of nodes 1, 2, 3, 4, 5 get the low trust value about these 5 nodes. The three nodes trust value is the lowest. Therefore who can determine the node according to the trust value can easily identify a malicious node.

3.4. Reduce Packet Loss Rate. 100 nodes are randomly scattered in the area of 100 m ∗ 100 m. Each over a certain time, the node randomly selects a node as a destination node, and sending data packets in accordance with the improved model. Trust node packet loss rate of 0.01. Malicious node gray hole attacks, and packet loss rate is set to 0.65–0.75.

Figure 6 shows the malicious node in the network by a packet loss rate becomes 10 when, from the figure, we can see that when the packet loss rate is still relatively low, about 0.05, at a time when network in the number of malicious nodes has accounted for 20%.

Figure 7 shows the network of malicious nodes gradually increasing to 10% packet loss rate can be seen from the figure the malicious nodes in the network to reach 30%, and the packet loss rate of the network remains a low state. Thus, our improved model to ensure that the node transmits the packet path is a secure path, thereby reducing the probability of packet loss of the node.

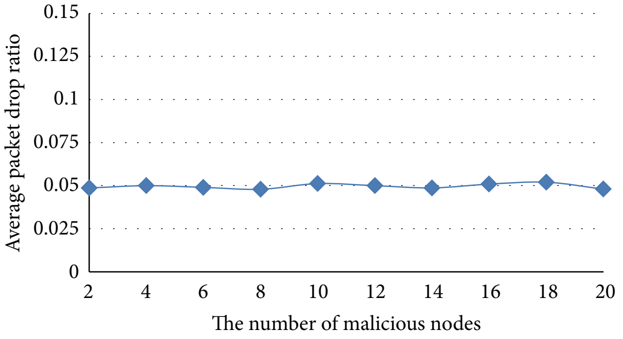


Figure 6: Network package drop ratio (with number of nodes from 2 to 20).

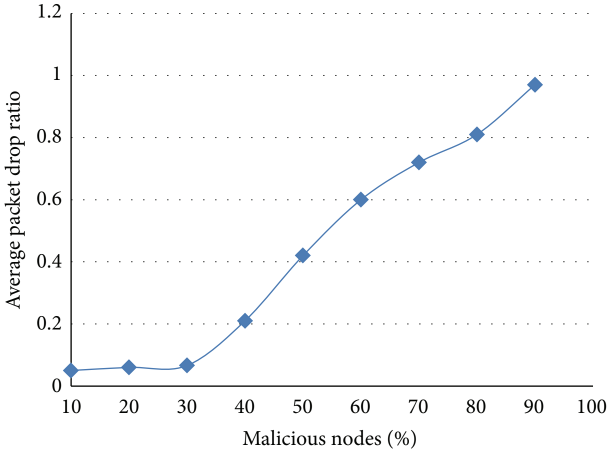


Figure 7: Network package drop ratio (with more nodes).

3.5. Average of Path Length. Next, let 100 nodes randomly scattered in the area of 100 m ∗ 100 m and use 5 different topologies for conducting simulation. The sink node is also randomly selected and the data packets were sent and forwarded according to the improved model. After the simulation, the average of the path length is counted and depicted in Figure 8.

In Figure 8, the red line indicates the average of path length in the case without malicious nodes and we take this as the baseline. From this figure, we can see that the average of path length differs slightly from the baseline along the increasing of the number of malicious nodes. That is, the average of path length is controlled within 1.5 hops. This suggests that the improved model does not infer much resource consuming toward each node, and the additional overhead is acceptable.

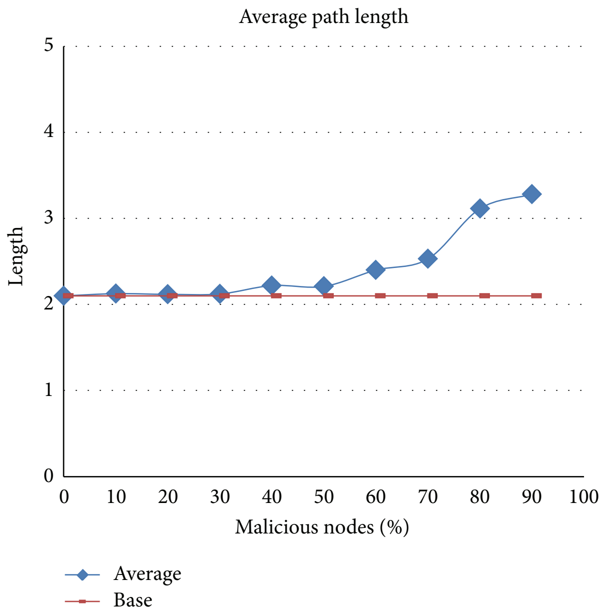


Figure 8: Average of path length.

4. Application Cluster Head Election

The cluster head is the basis for cluster formation. Once a malicious node has been a cluster head, the consequences would be disastrous. As far as we know, the cluster routing protocols are largely based on this assumption: all wireless sensor nodes are trustworthy [5]. This assumption may naturally lead the malicious or compromised node to be selected as the cluster head, which could be a threat to the network. Therefore, we need an effective mechanism to identify the captured nodes and ensure the cluster head is trustworthy.

4.1. Cluster Head Election Based on New Trust Model. The cluster head election can be divided into two parts. One is the nodes updating their trust values; the other is the cluster head election. In the first part, nodes use modified BTRM-WSN to find the most trusted paths leading to the cluster head. During this process, the trust value of each neighbor is updated as well. In the second part, we refer to Garth's mechanism [5, 7] to elect trustworthy cluster head. In this section, our framework will be introduced in two processes: cluster head and member nodes.

The flow chart of cluster head is shown in Figure 9. The major blocks are explained in details as follows.

(1) When the current cluster head's battery power level falls below a predetermined threshold or serve for a predetermined period of time, it broadcasts (within the cluster) a new election message.

(2) After cluster members vote, the current cluster head then tallies the votes and decides the winner based on simple majority. The node with the second highest number of votes is selected as the vice cluster head. The purpose of the vice cluster head is to assume cluster head function in the event that the newly elected cluster head fails before handing over to its successor.

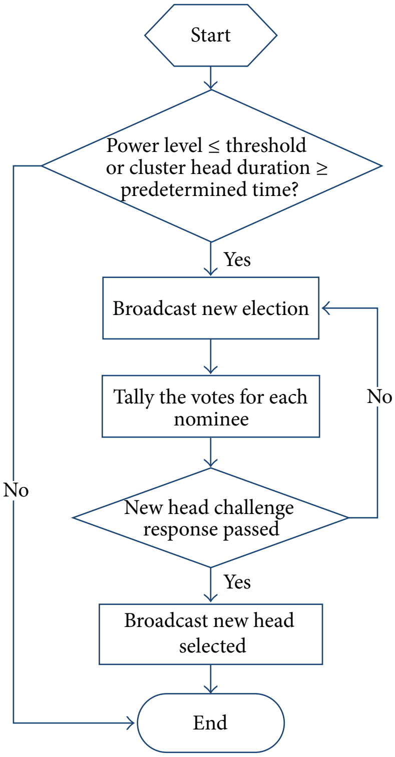

Start
Power level ≤ threshold or cluster head duration ≥ predetermined time?
Yes
No
Broadcast new election
Tally the votes for each nominee
New head challenge response passed
No
Yes
Broadcast new head selected
End

Figure 9: Flow chart of cluster head.

(3) The new winner and the vice cluster head have to pass a challenge response from the current cluster head before they are allowed to take up office.

(4) If one or both of them fail, the incumbent cluster head informs the cluster members and initiates a new election.

(5) If they success, the cluster head multicasts the winner and runner-up to all the members of the cluster.

The flow chart of cluster members is shown in Figure 10. The major blocks are explained in details as follows.

(1) All the members use the improved BTRM-WSN to find the most trusted path leading to the cluster head and send their data to cluster head. When they receive a notification about cluster head election, all nodes vote for a new cluster head.

(2) They select a candidate node with the highest trust value from their trust table.

(3) Then, they send votes to current cluster head. For greater security, the vote is encrypted. Neighbors therefore have no idea of the vote content of each other.

(4) After the members receive a broadcast about new cluster head, they now communicate with the new cluster head.

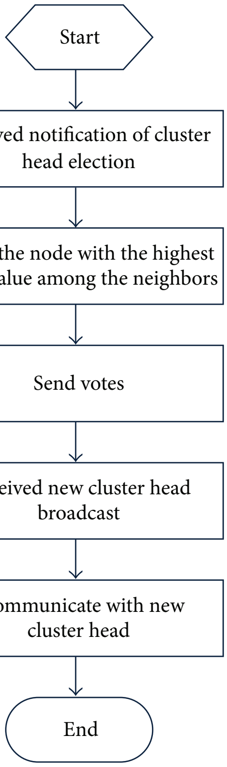

Start
Received notification of cluster head election
Select the node with the highest trust value among the neighbors
Send votes
Received new cluster head broadcast
Communicate with new cluster head
End

Figure 10: Flow chart of member nodes.

4.2. Evaluation. In this section, we evaluate the performance of our framework in two aspects: (1) the capability in preventing compromised nodes form being selected as the cluster head; and (2) the standard deviation shows the average probability of selecting compromised nodes. We use TRMSim-WSN [6] as our main simulation platform.

During our experiments, we use a flat, rectangular area of 100 m $*$ 100 m. 50 nodes are randomly deployed and formed as one cluster. The nodes transmission distance is 18 meters. We launched our model 300 times over 5 random WSNs. Every 10 times, we initiate a new cluster head election.

4.2.1. Probability of Selecting Compromised Nodes. We increase the number of malicious nodes from 10% gradually to 90% to test the robustness of our model. The good nodes run our cluster head election algorithm, while the malicious ones pick up a candidate randomly from their neighbors. We omit the challenge response procedure to avoid complicating this simple system.

Figure 11 shows the average probability. For clusters with less than 30% of compromised nodes, our mechanism almost never selects a compromised node. However, the probability increases rapidly after 70% of the nodes were compromised. This can be explained that malicious nodes send false votes, which greatly interferes with the result of election. Figure 12 shows the comparison with Crosby's model [5]. A conclusion that can be obtained is that the accuracy rate of these two models, on preventing the malicious or compromised nodes from being a cluster head, is basically the same. This also demonstrates the effectiveness of our framework in securing cluster.

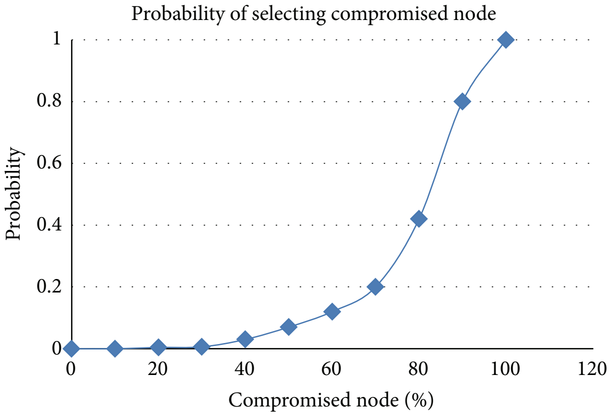


Figure 11: Probability of selecting compromised nodes.

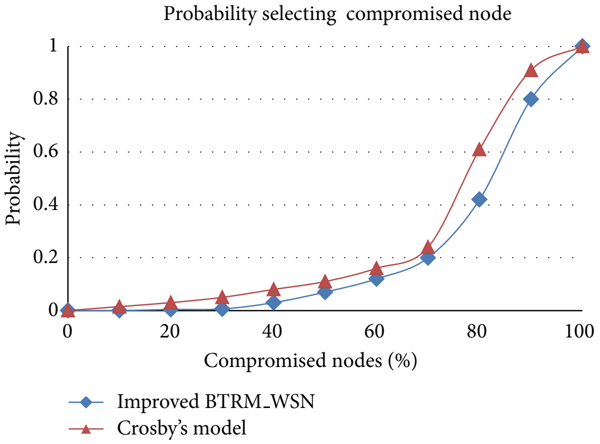


Figure 12: Comparison with Crosby's model.

4.2.2. Standard Deviation. Figure 11 actually shows the average probability of selecting compromised nodes in our framework over 5 random WSNs. But an average probability 0.8, for instance, could be reached because the model always selected a trustworthy cluster head on probability 0.8, or just because it selected on 1.0 in half of the tested wireless sensor networks and 0.6 in the other half.

This is the reason why we decided to measure and show the standard deviation related to that average as well. Figure 13 shows the results.

We can see the standard deviation also remains quite low. This means our framework is able to select a trustworthy cluster head with a quite high accuracy, regardless the topology of the WSNs.

5. Conclusion

In this paper, we propose a new trust model by making several improvements toward Marmol et al.'s BTRM-WSN model: adopting an interactive multiple ant colony algorithm,

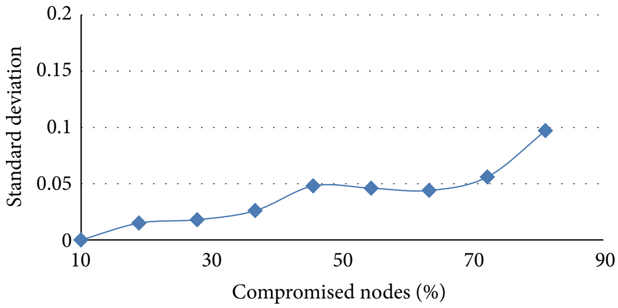


Figure 13: Standard deviation.

introducing new node types and more reasonable meta-assumption, new trust value evaluation function and penalty function, and so forth. The resulted model is more efficient and suited for general WSN-oriented application scenarios. In particular, the new model is very effective in identifying malicious nodes and decreasing packet loss rate according to simulations. As a typical application on our proposal, we also present a cluster head election framework based on interactive ant colony algorithm for WSNs. Further simulations show that our framework is feasible and has high accuracy in preventing compromised nodes from being a cluster head. In the future work, we will examine the scalability of our model through comprehensive simulations and try to integrate other trust models into our framework.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC) (no. 61103198) and the Engineering Science Program of Communication University of China (no. 3132013XNG1319).

References

[1] F.-Y. Ren, H.-N. Huang, and C. Lin, "Wireless sensor networks," *Journal of Software*, vol. 14, no. 7, pp. 1282–1291, 2003 (Chinese).

[2] J. Hurt, Y. Lee, H. Yoont, D. Choi, and S. Jin, "Trust evaluation model for wireless sensor networks," in *Proceedings of the 7th International Conference on Advanced Communication Technology (ICACT '05)*, pp. 491–496, Phoenix Park, Republic of Korea, February 2005.

[3] Q. Jing, L. Y. Tang, and Z. Chen, "Trust management in wireless sensor networks," *Journal of Software*, vol. 19, no. 7, pp. 1716–1730, 2008.

[4] J. Hur, Y. Lee, S. Hong et al., "Trust-based secure aggregation wireless sensor networks," in *Proceedings of the 3rd International Conference on Computing, Communications and Control Technologies*, vol. 3, pp. 1–6, 2005.

[5] G. V. Crosby, N. Pissinou, and J. Gadze, "A framework for trust-based cluster head election in wireless sensor networks," in *Proceedings of the 2nd IEEE Workshop on Dependability and Security in Sensor Networks and Systems (DSSNS '06)*, pp. 13–22, IEEE Computer Society, Columbia, Md, USA, April 2006.

[6] Y. L. Sun, W. Yu, Z. Han, and K. J. R. Liu, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 305–317, 2006.

[7] F. G. Mármol and G. M. Pérez, "Providing trusting wireless sensor networks using a bio-inspired technique," *Telecommunication Systems Journal*, vol. 46, no. 2, pp. 163–180, 2011.

[8] A. Aljanaby, K. R. Ku-Mahamud, and N. M. Norwawi, "Interacted multiple ant colonies optimization approach to enhance the performance of ant colony optimization algorithms," *Computer and Iformation Science*, vol. 3, no. 1, 2010.

[9] F. G. Mármol and G. M. Pérez, "TRMSim-WSN, trust and reputation models simulator for wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications, Communication and Information Systems Security Symposium*, June 2009.

[10] T. K. Kim and H. S. Seo, "A Trust Model using fuzzy logic in wireless sensor networks," *Proceedings of World Academy of Science, Engineering and Technology*, vol. 32, 2008.

[11] B. Shen, S.-Y. Zhang, and Y.-P. Zhong, "Cluster-based routing protocols for wireless sensor networks," *Journal of Software*, vol. 17, no. 7, pp. 1588–1600, 2006.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 948641, 9 pages
http://dx.doi.org/10.1155/2013/948641

Research Article

Trust Management Scheme Based on D-S Evidence Theory for Wireless Sensor Networks

Renjian Feng,[1] Shenyun Che,[1] Xiao Wang,[1,2] and Ning Yu[1]

[1] *School of Instrumentation Science and Opto-Electronics Engineering, Beihang University, Beijing 100191, China*
[2] *Science and Technology on Communication Information Security Control Laboratory, Jiaxing 314033, China*

Correspondence should be addressed to Renjian Feng; rjfeng@buaa.edu.cn

Received 9 January 2013; Revised 29 May 2013; Accepted 5 June 2013

Academic Editor: Anfeng Liu

Trust management scheme has been regarded as a powerful tool to defend against the wide set of security attacks and identify malicious nodes. In this paper, we propose a trust management scheme based on revised Dempster-Shafer (D-S) evidence theory. D-S theory is preponderant in tackling both random and subjective uncertainty in the trust mechanism. A trust propagation mechanism including conditional trust transitivity and dynamic recommendation aggregation is developed for obtaining the recommended trust values from third part nodes. We adopt a flexible synthesis method that uses recommended trust only when no direct trust exists to keep a good trust-energy consumption balance. We also consider on-off attack and bad mouthing attack in our simulation. The simulation results and analysis show that the proposed method has excellent ability to deal with typical network attacks, better security, and longer network lifetime.

1. Introduction

Wireless sensor networks (WSNs) consist of plentiful tiny, sensing capabilities, and resource-constrained sensor nodes, and are often deployed in unattended and hostile environments to perform various monitoring tasks [1, 2]. However, due to the wireless and unattended deployment nature of WSNs, there is a risk of unique threats [3]. Hence, security plays a vital role in guaranteeing the normal running of the whole network. Although security requirements of WSNs are quite similar with those of conventional networks, the security strategies based on the traditional authentication and encryption mechanisms are unsuitable to apply to WSNs because of the nodes' resource constraints [4]. Therefore, the trust management scheme has attracted more and more research attentions as a complementary security mechanism [5]. The basic idea of the trust management scheme is to calculate trust values that are used to describe the trustworthiness, reliability, or competence of individual nodes, based on some monitoring schemes [6]. Then the trust information can be applied to higher layer decisions such as routing [7, 8], data aggregation [9], and cluster head election [10, 11]. To the best of our knowledge, a number of trust management schemes have been proposed for WSNs [12–22], but most of them failed to establish a reasonable trust management scheme to express the subjectivity, uncertainty, and transitivity of trust characteristics in WSNs.

To resolve the problems, this paper puts forward a trust management scheme (TMS) based on revised D-S evidence theory in WSNs and achieves main contributions as follows. (1) A trust propagation mechanism including conditional trust transitivity and dynamic recommendation aggregation using the revised D-S evidence theory is proposed, which maintains the subjectivity, uncertainty, and transitivity of trust characteristics. (2) An adaptive time factor is adopted to dynamically weight history experience against current information, which enhances the accuracy of trust calculation. (3) To keep a good trust-energy consumption balance, a synthesis method that uses recommended trust only when no direct trust exists is proposed. (4) We address the issue of TMS performance in terms of ability to defeat some attacks (on-off attack, bad mouthing attack), detection of malicious nodes, and energy consumption, comparing with NBBTE [15] and BRSN [16]. Simulation results demonstrate that TMS has excellent ability to deal with typical network attacks, better security, and longer network lifetime.

The model proposed in this work extends our prior work [15] which integrated the approach of nodes behavioral strategies and modified evidence theory. In this paper, we improve the previous model with mechanisms for the propagation of nodes' recommendation and the synthesis of nodes' trust value. Moreover, we refine the algorithm of direct trust value, evaluate our scheme's ability to defeat on-off attack and bad-mouthing attack, and study the security and energy consumption of the model.

The rest of this paper is organized as follows. Section 2 presents related work on trust establishment for WSNs. Section 3 describes the D-S evidence theory and the process of TMS, including computation of nodes' trust value. In Section 4, comparing with NBBTE and BRSN, the superiority of TMS is shown by simulations. Finally, the conclusions are presented in Section 5.

2. Related Works

The research on establishing trusts can be classified into two categories, reputation-based [16–19] and trust establishment [20–22]. In the former category, trust is evaluated by direct observation and second-hand information distributed among a network. In the latter category, trust in neighbors is evaluated by direct observation and trust relations between two nodes.

Reputation-based framework for sensor networks (RFSN) [16] used watchdog mechanism to build trust rating. Within the framework of RFSN, a beta reputation system for sensor networks (BRSN) that used a Bayesian formulation was employed. Since then, many researches have been done based on the BRSN model such as MA&TP-BRSN, and RFM-WSN [17]. However, in RFSN, the stipulation that no node is allowed to disseminate bad reputation information makes it unable to cope with uncertain situations. Aivaloglou and Gritzalis [18] proposed a hybrid trust and reputation management protocol by exploiting the predeployment knowledge on the network topology and the information flows. But it is not easy to get the predeployment knowledge. In [19], the authors proposed a behavior reputation method which defined the similarity and the similarity matrix by using normal differences of the status estimate vectors. However, the initialization stage of the model is based on the authentication key which is prone to attacks.

Zarei et al. [20] presented a novel congestion control scheme based on fuzzy logic systems. The proposed scheme enabled the nodes to investigate the behavior of their neighbors and isolated them upon malfunctioning, decreasing congestion problem, and buffering capacity shortage. However, the use of fuzzy logic makes it easy to lose some information and may lead to an inaccurate result. In [21], the authors proposed a new lightweight group-based trust management scheme. In this model, each sensor node (SN) performed peer evaluation based on direct observations or recommendations, and each cluster head (CH) evaluated other CHs as well as SNs under its own cluster. However, trust in their case is assessed only based on past interaction experiences in message delivery. Lopez et al. [22] listed the best practices that were essential for developing a good trust management system and made an analysis of the state of the art related to these practices. The reference makes an excellent summary, proposes many profound viewpoints, and shows an additional insight on the trust evaluation field.

3. TMS Algorithm

Refer to [4], we define trust as the confidence that node i (denoted as n_i) has on node j (denoted as n_j) about how n_j will perform as expected. A complete trustworthiness consists of subject entity's observation and recommendation from third party. The TMS algorithm firstly establishes various trust factors based on our previous work [15]. Next, direct trust is calculated on the base of trust factors. Then, the recommendations of several neighbor nodes are acquired in accordance with the revised D-S rule and the trust difference between pieces of evidence. Finally, the overall trust value is computed through a flexible synthesis method that guarantees a good trust-energy consumption balance. Figure 1 shows the structure of TMS algorithm.

3.1. D-S Evidence Theory. Due to the subjectivity of trust evaluation, it is unsuitable to simply establish the recommended trust value by weighted average. D-S evidence theory can briefly express the important conceptions, such as "uncertainty," and make right judgments by efficiently integrating many-sided uncertain information. Hence, in our proposed algorithm, we calculate trust value and the average weight of recommendations based on the D-S rule. The basic definitions of D-S theory are defined as follows [23].

Definition 1. Let Ω be the identification frame, denoting a set of mutually exclusive and exhaustive hypotheses about problem domains. Correspondingly, 2^{Ω} is the power set of Ω.

Definition 2. Mass stands for a belief mapping from 2^{Ω} to the interval between 0 and 1, represented as $m. m : 2^{\Omega} \rightarrow [0,1]$ is called the BPA (Basic Probability Assignment) and is defined as below:

$$\begin{gathered} m(\varnothing) = 0, \\ \sum_{A \subseteq \Omega} m(A) = 1, \quad A \neq \varnothing. \end{gathered} \tag{1}$$

Definition 3. The belief of a hypothesis is the sum of the beliefs for those hypotheses that are its subsets. Its definition is given as

$$\operatorname{Bel}(A) = \sum_{B \subseteq A} m(B), \quad \forall A \subseteq \Omega, \tag{2}$$

where A is named focal element and $m(A) > 0$ is the basic confidence level of A, representing how much the evidence supports A to happen.

3.2. Trust Factors. To defeat various attacks, we had better take all kinds of factors that depend on the interactions between neighbor nodes into consideration. However, there is an obvious trade-off between the number of factors and the

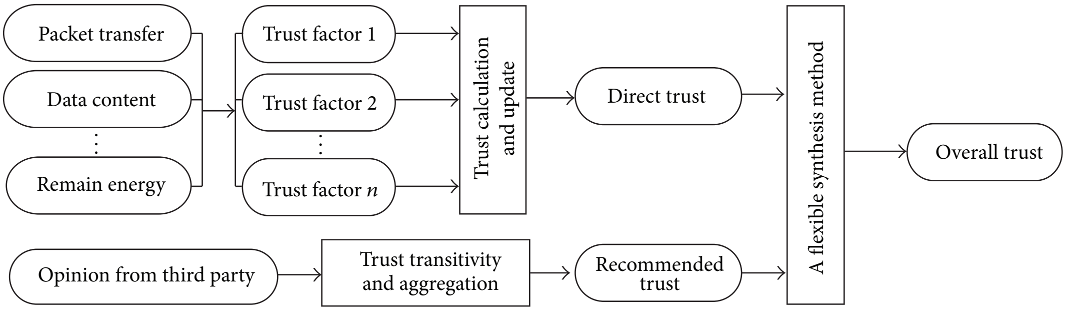


Figure 1: The structure of TMS algorithm.

energy consumption. We select four trust factors from our previous work [15]. Suppose n_i evaluates the trust degree on n_j; the trust factors are Received Packets Rate $RPF_{i,j}(t)$, Successfully Sending Packets Rate $SPF_{i,j}(t)$, Packets Forwarding Rate $TPF_{i,j}(t)$, and Node Availability $HPF_{i,j}(t)$.

3.3. Direct Trust Evaluation Approach. Subject n_i monitors the behaviors of object n_j in one cycle and acquires the current trust value $CDT_{i,j}(t)$ based on the following expression:

$$\begin{aligned} m_{i,j}^{C}(\{T\}) &= f_1\left(RPF_{i,j}, SPF_{i,j}, TPF_{i,j}, HPF_{i,j}\right), \\ m_{i,j}^{C}(\{-T\}) &= f_2\left(RPF_{i,j}, SPF_{i,j}, TPF_{i,j}, HPF_{i,j}\right), \\ m_{i,j}^{C}(\{T,-T\}) &= 1 - m_{i,j}^{C}(\{T\}) - m_{i,j}^{C}(\{-T\}). \end{aligned} \tag{3}$$

The functions f_1 and f_2 are chosen in advance according to the specific assignments of network.

Furthermore, the direct trust value is recalculated in accordance with history records. The update of direct trust value is calculated as follows:

$$DT_{i,j} = \beta \times HDT_{i,j} + (1-\beta) \times CDT_{i,j}, \tag{4}$$

where $DT_{i,j}$ is the direct trust value of subject n_i on object n_j in current cycle; $HDT_{i,j}$ is the direct trust value of latest cycle; parameter β is the adaptive time factor used to weight history experience against current information. To keep β preferably dynamic, it is satisfied as follows:

$$\beta = \begin{cases} \beta_s, & m_{i,j}^{H}(\{T\}) \geq m_{i,j}^{C}(\{T\}), \\ \beta_l, & m_{i,j}^{H}(\{T\}) < m_{i,j}^{C}(\{T\}), \end{cases} \tag{5}$$

where $0 < \beta_s < \beta_l < 1$. The parameter $m_{i,j}^{C}(\{T\})$ and $m_{i,j}^{H}(\{T\})$ represent the trust components of $CDT_{i,j}$ and $HDT_{i,j}$, respectively.

3.4. Recommended Trust Evaluation Approach

3.4.1. Trust Transitivity. Suppose the recommended trust value of n_i on n_j can be obtained through s different paths. And the number of recommendation paths s depends on nodes' distribution and transmission radius. In order to avoid trust recycle recursion and decrease network communication payload, the recommendation values are confined to direct trust value of the common neighbors owned by both n_i and n_j. As shown in Figure 2, n_i can only get the trust recommendation of n_j from $k_1, k_2, k_3 \ldots, k_s$.

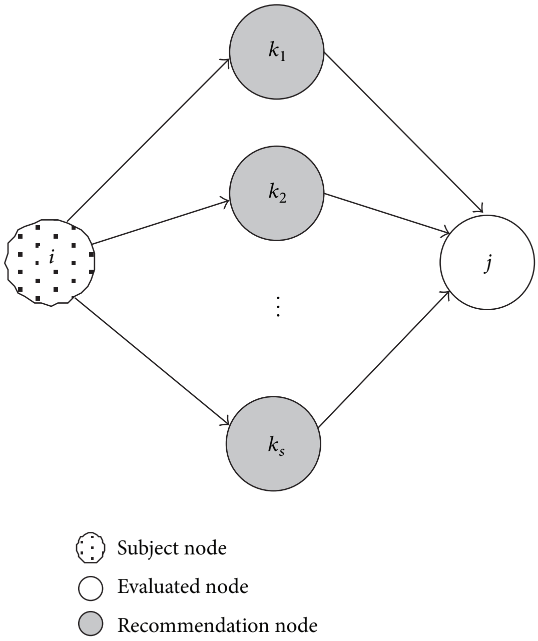


Figure 2: Recommendation relationship between subject n_i and object n_j.

Assume that $RT_{i,j}^{1}$ is the recommended trust value of n_i on n_j through recommendation path $pt1 = \{k1\}$. The vector forms of $RT_{i,j}^{1}$, DT_{i,k_1}, $DT_{k_1,j}$ are as follows:

$$\begin{aligned} RT_{i,j}^{1} &= \left(m_{i,j}^{1}(\{T\}), m_{i,j}^{1}(\{T,-T\}), m_{i,j}^{1}(\{-T\})\right), \\ DT_{i,k_1} &= \left(m_{i,k_1}^{D}(\{T\}), m_{i,k_1}^{D}(\{T,-T\}), m_{i,k_1}^{D}(\{-T\})\right), \\ DT_{k_1,j} &= \left(m_{k_1,j}^{D}(\{T\}), m_{k_1,j}^{D}(\{T,-T\}), m_{k_1,j}^{D}(\{-T\})\right). \end{aligned} \tag{6}$$

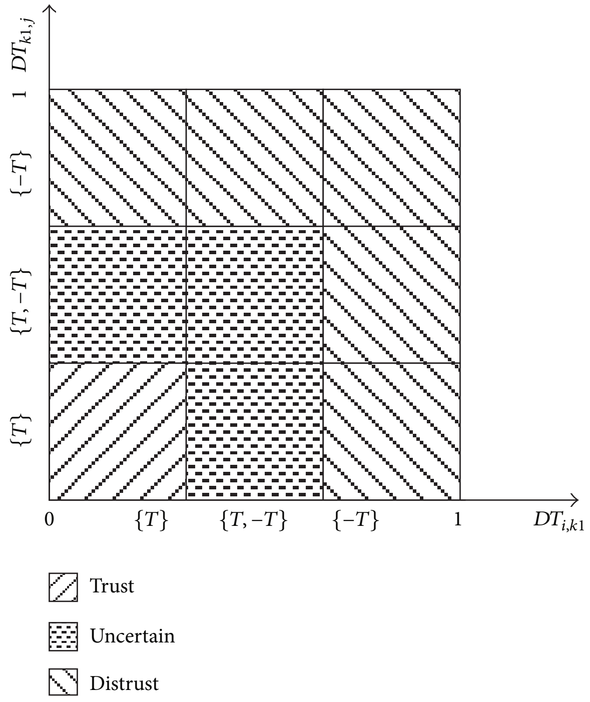


FIGURE 3: The process of trust transitivity.

Let us set $\theta = \{\{T\},\{T,-T\},\{-T\}\}$, A, E and $F \subseteq \theta$. Then, the $RT^{1}_{i,j}$ is calculated as

$$m^{1}_{i,j}(A) = \begin{cases} m^{D}_{i,k_1}(A) \times m^{D}_{k_1,j}(A), & A = \{T\} \\ \displaystyle\sum_{E=A \text{ or } F=A} m^{D}_{i,k_1}(E) \times m^{D}_{k_1,j}(F), & A = \{-T\} \\ 1 - m^{1}_{i,j}(\{T\}) - m^{1}_{i,j}(\{-T\}), & A = \{T,-T\}. \end{cases} \quad (7)$$

Using the symbol ⊗ to denote this operation, we can get

$$RT^{1}_{i,j} = DT_{i,k_1} \otimes DT_{k_1,j}. \quad (8)$$

To vividly show the process of trust transitivity, we resort to Figure 3. It is obvious to see that as long as one of DT_{i,k_1} and $DT_{k_1,j}$ is distrust, $RT^{1}_{i,j}$ is distrust.

Extending the above transitivity to multihop, we can get recommended trust through complex recommendation paths with many middle nodes as follows:

$$RT^{1}_{i,j} = DT_{i,\bullet} \otimes \cdots \otimes DT_{\bullet,j}, \quad (9)$$

where the symbol • indicates anonymous nodes in recommendation paths.

3.4.2. Dynamic Aggregation of Recommended Trust. On the basis of trust transitivity, n_i obtains recommended trust values on n_j through s recommendation paths; namely:

$$\begin{aligned} RT^{1}_{i,j} &= \left(m^{1}_{i,j}(\{T\}), m^{1}_{i,j}(\{T,-T\}), m^{1}_{i,j}(\{-T\})\right), \\ RT^{2}_{i,j} &= \left(m^{2}_{i,j}(\{T\}), m^{2}_{i,j}(\{T,-T\}), m^{2}_{i,j}(\{-T\})\right), \\ &\vdots \\ RT^{s}_{i,j} &= \left(m^{s}_{i,j}(\{T\}), m^{s}_{i,j}(\{T,-T\}), m^{s}_{i,j}(\{-T\})\right). \end{aligned} \quad (10)$$

Then, n_i would aggregate these pieces of evidence to get a consensus on n_j. Due to the existence of malicious nodes that may offer false recommendations, we introduce the revised D-S combination rule which adopts a consistent intensity to adjust weights of recommended trust values. The integration process is described in detail as follows.

Firstly, we compute the corresponding average weight denoted as I_u. The consistent intensity between $RT^{u}_{i,j}$ and $RT^{v}_{i,j}$ is defined as

$$I_{u,v} = 1 - \sqrt{\frac{1}{2}\left(\left\|\vec{m}^{v}_{i,j}\right\|^{2} + \left\|\vec{m}^{u}_{i,j}\right\|^{2} - 2\left\langle \vec{m}^{v}_{i,j}, \vec{m}^{u}_{i,j}\right\rangle\right)}, \quad v = 1, 2 \ldots, s; \quad u = 1, 2 \ldots s, \quad (11)$$

where $\|\vec{m}^{v}_{i,j}\|^{2} = \langle \vec{m}^{v}_{i,j}, \vec{m}^{v}_{i,j}\rangle$, $\|\vec{m}^{u}_{i,j}\|^{2} = \langle \vec{m}^{u}_{i,j}, \vec{m}^{u}_{i,j}\rangle$, $\langle \vec{m}^{v}_{i,j}, \vec{m}^{u}_{i,j}\rangle$ is the inner product of $\vec{m}^{v}_{i,j}$ and $\vec{m}^{u}_{i,j}$.

The difference between two recommended trust pieces of evidence increases with the reduction of consistent intensity. The lower the consistent intensity is, the more probably false trust recommendation may occur.

Furthermore, the matrix of consistent intensity which is composed of all the recommended trust values is defined as

$$I_{s\times s} = \begin{bmatrix} 1 & I_{1,2} & \cdots & I_{1,s} \\ I_{2,1} & 1 & \cdots & I_{2,s} \\ \vdots & \vdots & \ddots & \vdots \\ I_{s,1} & I_{s,2} & \cdots & 1 \end{bmatrix}. \quad (12)$$

Through summation in row and normalization, the totally consistent intensity of recommended trust $RT^{u}_{i,j}$, which is equal to the average weight I_u, is computed by

$$I_u = \frac{\sum_{v=1, v\neq u}^{s} I_{u,v}}{\mathrm{Max}\left(\sum_{v=1, v\neq w, 1\leq w\leq s}^{s} I_{w,v}\right)}. \quad (13)$$

Then, the basic reliability function m of every recommended trust evidence is amended by I_u as follows:

$$\begin{gathered} m^{u\prime}_{i,j}(\{T\}) = I_u \times m^{u}_{i,j}(\{T\}), \\ m^{u\prime}_{i,j}(\{-T\}) = I_u \times m^{u}_{i,j}(\{-T\}), \\ m^{u\prime}_{i,j}(\{T,-T\}) = 1 - m^{u\prime}_{i,j}(\{T\}) - m^{u\prime}_{i,j}(\{-T\}), \\ u = 1, 2, \ldots, s. \end{gathered} \quad (14)$$

Above all, the recommended trust can be modified as

$$RT^{u}_{i,j} = \left(m^{u\prime}_{i,j}(\{T\}), m^{u\prime}_{i,j}(\{T,-T\}), m^{u\prime}_{i,j}(\{-T\})\right), \quad u = 1, 2, \ldots, s. \tag{15}$$

Finally, we can get the consistent recommended trust $RT'_{i,j} = (m'_{i,j}(\{T\}), m'_{i,j}(\{T,-T\}), m'_{i,j}(\{-T\}))$ as follows:

$$m'_{i,j}(\{A\}) = \frac{\sum_{u=1}^{u=s} m^{u\prime}_{i,j}(\{A\})}{\sum_{u=1}^{u=s} I_u}, \qquad A = \{T\}, \quad A = \{-T, T\}, \quad A = \{-T\}. \tag{16}$$

3.5. Synthesis of Overall Trust Value. The recommendation trust is useful to get a more accurate trust value, but calculating it will consume more energy. Thus there is a need for a good trust-energy consumption balance in the trust management system. To solve this problem, we calculate the overall trust by a flexible synthesis method which works as follows: only when n_i does not have direct evidence on n_j, the recommendation trust is taken into account. Hence, the overall trust value $OT_{i,j}$ is

$$\begin{gathered} OT_{i,j} = \begin{cases} DT_{i,j} & \text{while node } i \text{ has direct evidence on node } j \\ RT'_{i,j} & \text{else} \end{cases} \\ OT_{i,j} = (m(\{T\}), m(\{-T, T\}), m(\{-T\})). \end{gathered} \tag{17}$$

If the decision model satisfies

$$\begin{gathered} m_{i,j}(\{T\}) - m_{i,j}(\{-T\}) > \varepsilon, \\ m_{i,j}(\{-T, T\}) < \theta, \\ m_{i,j}(\{T\}) > m_{i,j}(\{-T, T\}). \end{gathered} \tag{18}$$

Then subject n_i regards n_j as "Trust," and adds n_j into its trustworthiness list. In like manner, n_j can be marked "Uncertain" or "Distrust."

4. Simulation Results

In this section, we use Matlab platform to show TMS has better performance than NBBTE and BRSN in terms of ability to defeat some attacks (on-off attack, bad mouthing attack), detection of malicious nodes, and energy consumption.

4.1. Defense of Attacks

4.1.1. On-Off Attack. Trust is a dynamic event. A good entity may be captured by attackers and turns into compromise node. On the other side, an incompetent entity can redeem the way that its neighbors regard it and become competent due to environmental changes. Because of the nodes' resource limitation, some trust schemes adopted trust compensation mechanism. However, a smart attacker can capitalize on this feature of the trust schemes and create on-off attacks in which malicious entities behave well and badly alternatively [6]. To address this issue, we adopt the adaptive time factor β which is introduced in Section 3.2. β depends on specific situations. Here, we can choose $\beta_s = 0.3$, $\beta_l = 0.8$. In order to prevent the malicious node registering as a new user, the pessimistic initialization strategy of trust value is accepted. Suppose that malicious nodes cooperate well with neighbor nodes to get good trust records at the beginning of the simulation but behave badly after 40 rounds. The simulation results are shown in Figures 4 and 5.

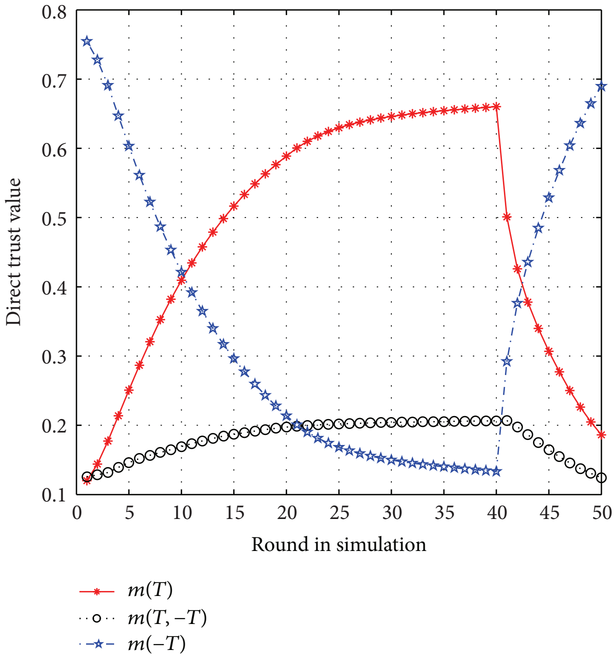


Figure 4: The change of direct trust value under on-off attack.

From Figure 4 we can see that $m(\{T\})$ increases slowly and $m(\{-T\})$ decreases slowly in the trust compensation stage (0–40 rounds). Once the malicious nodes behave badly, $m(\{T\})$ falls off sharply while $m(\{-T\})$ races up. In other words, the time for trust accumulation is much longer than that for trust collapse. It is because $\beta_l = 0.8$ which means that history information affects the trust value heavily in the trust compensation stage and $\beta_s = 0.3$ which means current information bulks large when attacks happen.

Figure 5 compares the trust value calculated by different methods under on-off attack. The trust value calculated by BFSN increases the fastest in the trust compensation stage and the trust value calculated by NBBTE has the slowest decline in the attacking stage. Both BFSN and NBBTE fail to resist on-off attack. On the contrary, TMS defends against on-off attack effectively as the trust value calculated by TMS has the slowest increase in the trust compensation stage and falls off sharply once the malicious nodes behave badly.

4.1.2. Bad Mouthing Attack. Once recommendations are taken into consideration, we take the risk of receiving

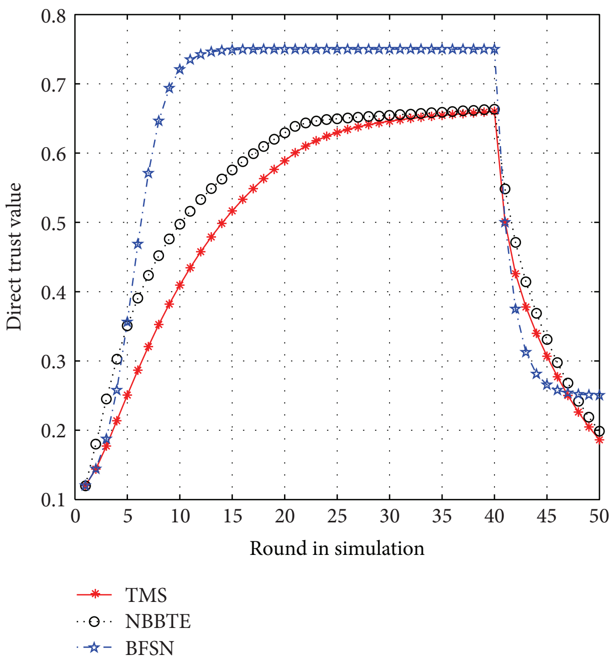


Figure 5: Comparison of trust value under on-off attack.

Table 1: Detailed information of twenty recommendations.

	$m(\{T\})$	$m(\{-T, T\})$	$m(\{-T\})$
RT^1	0.8061	0.0401	0.1538
RT^2	0.8162	0.0012	0.1826
RT^3	0.8461	0.0208	0.1331
RT^4	0.5152	0.0816	0.4032
RT^5	0.4952	0.0916	0.4132
RT^6	0.8132	0.0196	0.1672
RT^7	0.8262	0.0704	0.1034
RT^8	0.8035	0.0398	0.1567
RT^9	0.5092	0.0593	0.4315
RT^{10}	0.8137	0.0805	0.1058
RT^{11}	0.5002	0.0222	0.4776
RT^{12}	0.7975	0.0094	0.1931
RT^{13}	0.8071	0.0107	0.1822
RT^{14}	0.4971	0.0318	0.4711
RT^{15}	0.8186	0.0283	0.1531
RT^{16}	0.8279	0.0464	0.1257
RT^{17}	0.8182	0.0603	0.1215
RT^{18}	0.5538	0.0801	0.3661
RT^{19}	0.8072	0.0875	0.1053
RT^{20}	0.8123	0.0766	0.1111

dishonest recommendations which aim at framing good parties or boosting trust values of malicious peers [6]. This attack, referred to as the bad mouthing attack, is the most straightforward attack. Because of our flexible synthesis method, bad mouthing attack happens only when n_i has no direct evidence on n_j. To defeat this attack, we introduce the revised D-S rule that includes the average weight I_u to combine recommendation pieces of evidence.

Suppose n_i receives twenty recommendation pieces of evidence of credible n_j and RT^4, RT^5, RT^9, RT^{11}, RT^{14}, RT^{18} are false recommendation information. Refer to Table 1 for detailed information.

Combining those twenty pieces of evidence by our method, we can obtain $RT'_{i,j} = \{0.7327, 0.2288, 0.0385\}$. The average weight $I = (0.9997, 0.9893, 0.9805, 0.8772, 0.8630, 0.9968, 0.9788, 1.0000, 0.8657, 0.9806, 0.8374, 0.9921, 0.9942, 0.8408, 0.9968, 0.9895, 0.9900, 0.9007, 0.9783, 0.9842)$. It is obvious to see that I_4, I_5, I_9, I_{11}, I_{14}, I_{18} are smaller than others. Without the average weight, n_i would mistake n_j for unbelievable node.

To further explain TMS's ability to defeat against bad mouthing attack, we compare it with NBBTE and BFSN under two conditions: framing good parties and boosting trust values of malicious peers. The results are shown in Figures 6 and 7.

When a malicious node launches the bad mouthing attack which aims at framing good parties, BFSN performs excellent as it only propagates good reputation information about other nodes. However, it cannot prevent malicious nodes from boosting trust values of malicious peer, as shown in Figure 7. No matter which condition it is, TMS performs better than NBBTE. Considering that BFSN is incapable of dealing with the second condition, we can come to the conclusion that TMS defends against bad mouthing attack most effectively.

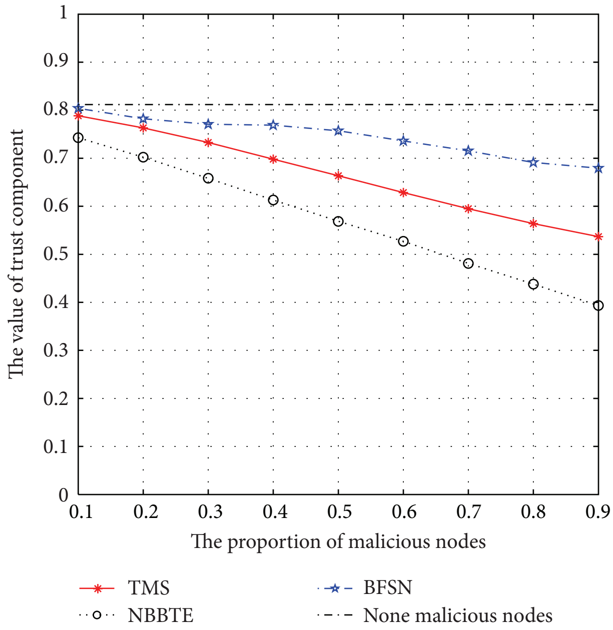


Figure 6: The trust value at different proportion of malicious nodes when framing good party.

4.2. Analysis of Network Security. To evaluate the network security, we compare our method with NBBTE and BRSN on the aspect of detecting malicious nodes. The proportions of

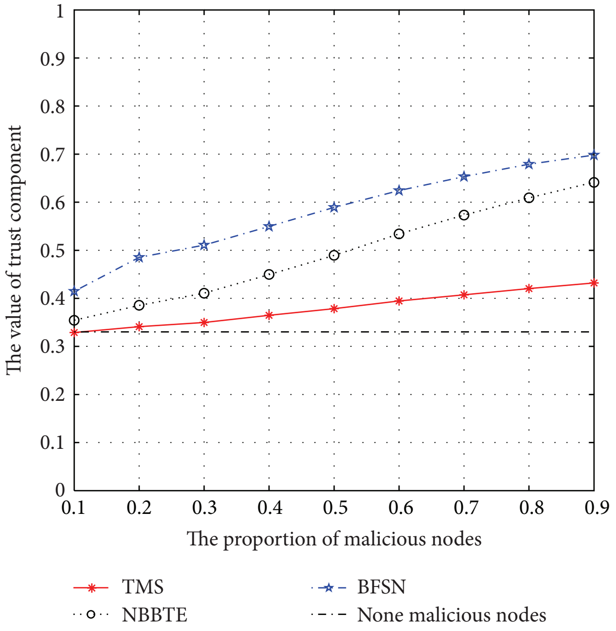


Figure 7: The trust value at different proportion of malicious nodes when boosting trust values of malicious peer.

detected malicious nodes under different trust mechanisms are shown in Figure 8.

It is obvious to see that TMS does better at detecting malicious nodes than BRSN. This results from two aspects. First, by using D-S theory, TMS takes subject uncertainty into consideration and avoids considering prior distribution, and consequently the accuracy of trust evaluation is improved. Second, we adopt the corresponding average weight of recommended trust, which increases the robustness of trust mechanism. The proportions of detected malicious nodes of TMS are little lower than that of NBBTE, because TMS uses recommended trust conditionally while BRSN considers both direct and recommended trust.

4.3. Analysis of Energy Consumption. To evaluate the performance of the flexible synthesis method proposed in Section 3.5, we make experiments on the energy consumption under different circumstances. The radio energy model proposed in [24] is used for our simulation. The simulation parameters are listed in Table 2 and the simulation results are shown in Table 3 and Figure 9.

Circumstance 1 is a special situation, where n_i has no neighbor. Compare circumstance 2 and circumstance 3, we can see that the more neighbors n_i has, the longer it will survive. It is because computing direct trust just needs one interaction while computing recommended trust needs two. Circumstance 3 and circumstance 4 tell us that the decreasing of average recommended pieces of evidence can increase n_i's lifetime. The reason is that decreasing one piece of average recommended evidence can reduce $2 \times (100 - a)$ interactions. In a word, the simulation results demonstrate that the flexible synthesis method saves energy greatly, especially when the number of average recommended pieces of evidence is small.

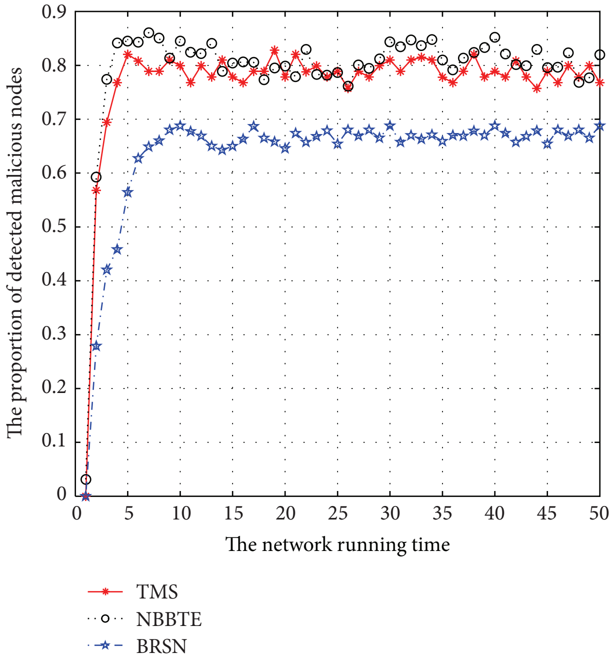


Figure 8: The proportions of detected malicious nodes under different trust mechanisms.

Table 2: Simulation parameters.

Parameters	Corresponding value
Simulation field	100 m × 100 m
Number of nodes	100
Transmission radius	15 m
Number of n_i's neighbor nodes	a
Number of average recommended pieces of evidence	b
Initial energy per node	0.5 J
Energy consumption of transmitter and receiver (E_{elec})	50 nJ/bit
Transmit Amplifier (ε_{amp})	100 pJ/bit/m^2
Message bits sended per round per node	4000 bit

Table 3: Round of n_i dies with different parameter values.

	a	b	Round of n_i dies
Circumstance 1	0	30	101
Circumstance 2	10	20	124
Circumstance 3	30	20	137
Circumstance 4	30	10	184

To further show how long n_i can survive by the flexible synthesis method of TMS comparing with NBBTE and BRSN, we count rounds that n_i can survive under different number of average recommended pieces of evidence. In this experiment, we set the number of n_i's neighbor nodes 30 and the number

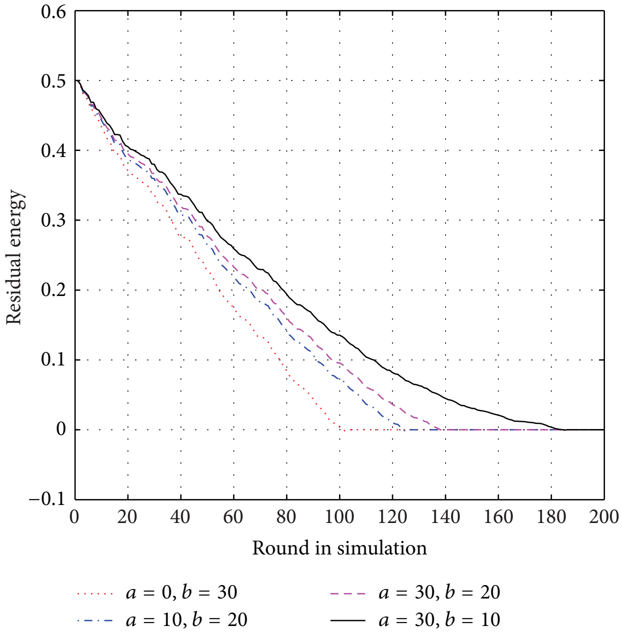


FIGURE 9: Residual energy of n_i under different circumstances.

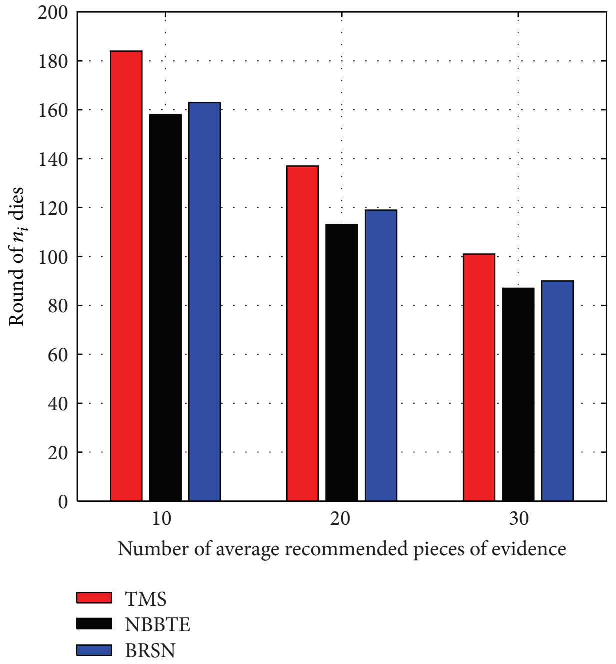


FIGURE 10: Round of n_i under different number of average recommended pieces of evidence.

of average recommended pieces of evidence 10, 20, and 30, respectively. The results are shown in Figure 10. It is obvious to see that n_i can survive the longest by using TMS.

5. Conclusions

In this paper, a trust management scheme (TMS) based on revised D-S evidence theory is proposed. It provides vector forms to express subjective trust opinions. On this basis, direct trust value on each neighbor node is calculated by considering trust factors which are defined according to node behaviors in order to detect malicious attacks. At the same time, recommended trust value from common neighbor nodes of subject and object nodes is obtained through conditional transitivity and the weight of each recommendation is obtained by revised D-S evidence theory. Afterwards, we use a flexible synthesis method to calculate the overall trust. Furthermore, the Matlab platform is used to test the performance of TMS, and simulation results show that the proposed algorithm can effectively resist vulnerabilities such as on-off attack and bad mouthing attack, reasonably evaluate trust levels of sensor nodes, and improve the network robustness and security. In addition, the flexible synthesis method saves energy greatly and, hence, prolongs the lifetime of WSNs.

Acknowledgments

The authors are grateful to the anonymous reviewers for their insightful comments. This work is supported by the National Natural Science Foundation of China under Grant no. 61201317 and no. 61001138.

References

[1] W. T. Zhu, J. Zhou, R. H. Deng, and F. Bao, “Detecting node replication attacks in wireless sensor networks: a survey,” *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1022–1034, 2012.

[2] M. Saleem, G. A. Di Caro, and M. Farooq, “Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions,” *Information Sciences*, vol. 181, no. 20, pp. 4597–4624, 2011.

[3] T. A. Zia, “Reputation-based trust management in wireless sensor networks,” in *Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP ’08)*, pp. 163–166, December 2008.

[4] P. Trakadas, S. Maniatis, P. Karkazis, T. Zahariadis, H. C. Leligou, and S. Voliotis, “A novel flexible trust management system for heterogeneous wireless sensor networks,” in *Proceedings of the International Symposium on Autonomous Decentralized Systems (ISADS ’09)*, pp. 369–374, March 2009.

[5] Y. Yu, K. Li, W. Zhou, and P. Li, “Trust mechanisms in wireless sensor networks: attack analysis and countermeasures,” *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.

[6] Y. L. Sun, Z. Han, and K. J. R. Liu, “Defense of trust management vulnerabilities in distributed networks,” *Communications Magazine*, vol. 46, no. 4, pp. 112–119, 2008.

[7] H. C. Leligou, P. Trakadas, S. Maniatis, P. Karkazis, and T. Zahariadis, “Combining trust with location information for routing in wireless sensor networks,” *Wireless Communications and Mobile Computing*, vol. 12, no. 12, pp. 1091–1103, 2012.

[8] H. Deng, Y. Yang, G. Jin, R. Xu, and W. Shi, “Building a trust-aware dynamic routing solution for wireless sensor networks,” in *Proceedings of the IEEE Globecom Workshops (GC ’10)*, pp. 153–157, December 2010.

[9] N. Poolsappasit and S. Madria, "A secure data aggregation based trust management approach for dealing with untrustworthy motes in sensor network," in *Proceedings of the 40th International Conference on Parallel Processing (ICPP '11)*, pp. 138–147, September 2011.

[10] R. J. Feng, S. Y. Che, and X. Wang, "A credible cluster-head election algorithm based on fuzzy logic in wireless sensor networks," *Journal of Computational Information Systems*, vol. 8, no. 15, pp. 6241–6248, 2012.

[11] G. V. Crosby, N. Pissinou, and J. Gadze, "A framework for trust-based cluster head election in wireless sensor networks," in *Proceedings of the 2nd IEEE Workshop on Dependability and Security in Sensor Networks and Systems (DSSNS '06)*, pp. 10–22, April 2006.

[12] W. R. Claycomb and D. Shin, "A novel node level security policy framework for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 418–428, 2011.

[13] F. Bao, I.-R. Chen, M. Chang, and J.-H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection," *IEEE Transactions on Network and Service Management*, vol. 9, no. 2, pp. 169–183, 2012.

[14] M. Momani, S. Challa, and R. Alhmouz, "BNWSN: bayesian network trust model for wireless sensor networks," in *Proceedings of the International Conference on Communications, Computers and Applications (MIC-CCA '08)*, pp. 110–115, August 2008.

[15] R. Feng, X. Xu, X. Zhou, and J. Wan, "A trust evaluation algorithm for wireless sensor networks based on node behaviors and D-S evidence theory," *Sensors*, vol. 11, no. 2, pp. 1345–1360, 2011.

[16] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, article 15, 2008.

[17] X. Gu, J. L. Qiu, and J. Wang, "Research on trust model of sensor nodes in WSNs," *Procedia Engineering*, vol. 29, pp. 909–913, 2012.

[18] E. Aivaloglou and S. Gritzalis, "Hybrid trust and reputation management for sensor networks," *Wireless Networks*, vol. 16, no. 5, pp. 1493–1510, 2010.

[19] M.-Z. Zhou, Y. Zhang, J. Wang, and S.-Y. Zhao, "A reputation model based on behavior trust in wireless sensor networks," in *Proceedings of the International Conference on Scalable Computing and Communications- 8th International Conference on Embedded Computing*, pp. 189–194, September 2009.

[20] M. Zarei, A. M. Rahmani, A. Sasan, and M. Teshnehlab, "Fuzzy based trust estimation for congestion control in wireless sensor networks," in *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems (INCoS '09)*, pp. 233–236, November 2009.

[21] R. A. Shaikh, H. Jameel, B. J. d'Auriol, H. Lee, S. Lee, and Y.-J. Song, "Group-based trust management scheme for clustered wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 11, pp. 1698–1712, 2009.

[22] J. Lopez, R. Roman, I. Agudo, and C. Fernandez-Gago, "Trust management systems for wireless sensor networks: best practices," *Computer Communications*, vol. 33, no. 9, pp. 1086–1093, 2010.

[23] C. Q. Tian and B. J. Yang, "A D-S evidence theory based fuzzy trust model in file-sharing P2P networks," *Peer-To-Peer Networking and Applications*, 2012.

[24] W. R. Heinzelman and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, pp. 3005–3014, 2000.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 912873, 16 pages
http://dx.doi.org/10.1155/2013/912873

Research Article

Low Mismatch Key Agreement Based on Wavelet-Transform Trend and Fuzzy Vault in Body Area Network

Yang Wu, Yongmei Sun, Lei Zhan, and Yuefeng Ji

State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Yang Wu; wuyang@bupt.edu.cn

Received 4 January 2013; Revised 15 April 2013; Accepted 10 May 2013

Academic Editor: Anfeng Liu

Body area network (BAN) is an emerging branch of wireless sensor networks for personalized applications in many fields, such as health monitoring. The services in BAN usually have a high requirement on security, especially for the medical diagnosis, which involves private information. With limitations of power and computation capabilities, one of the main challenges to ensure security in BAN is how to generate or distribute a shared key between nodes for lightweight symmetric cryptography. The current research almost exploits the randomness and distinctiveness (characteristics) of physiological signals to solve the key generation problem. However, it needs the help of additional hardware support and has the constraint on positions deployment, to acquire vital signals, which will bring the high cost and hardness to implementation of real system. To avoid the above problems, this paper presents a novel key generation scheme and a key distribution protocol, both of which are only based on wireless modules equipped on sensors. By exploiting the high correlation of received signal strength index (RSSI) between peer-to-peer communications, our scheme can provide a shared symmetric cryptographic key under the presence of an eavesdropper. We conduct experiments on the real Telosb nodes, and the results demonstrate that our proposed methods have a good performance on security.

1. Introduction

With the aged tendency of global population, people pay more attention to the healthcare requirements. Due to the advances of wireless communication and medical sensing technologies, many researchers focus on the applications of body area network (BAN), which can provide individual healthcare services. BANs consist of small and intelligent wireless medical sensors, worn or implanted in the human bodies, to sample vital physiological signals and send the records to a basestation (usually a portable device such as PDA or mobile phone) for real-time analysis or remote diagnosis. When a disaster or ailment comes up, BANs could offer an emergency response. Recently, some BAN platforms have been put into market. Fujitsu has developed inductively powered ring sensor [1], which works in conjunction with Fujitsu's developing sensor network system to tie into a healthcare monitoring of patients application for Apple's iPhone. Shimmer [2] has designed a wearable sensor platform to monitor a subject's EMG, ECG, and GSR. Due to the strong privacy and liability concerns of health data, these devices and communications between them need to be secured. "Lack of adequate security features may not only lead to a breach of patient privacy, but also potentially allow adversaries to compromise patient safety by modifying actual data resulting in wrong diagnosis and treatment [3]."

Given limited power and computation capabilities of BAN, asymmetric cryptography is not suitable. The only left option is symmetric encryption, which however has a significant challenge in dynamically sharing a secret key between devices (both sensors and base station). There are two traditional approaches to address the problem: predeployment or a key management infrastructure (a trusted third party), written in IEEE 802.15.6 standard for body area networks [4]. As to predeployment, if the initial key is preconfigured in hardware by manufactures, sensors from different companies could not work together. When using something like SD card or USB stick instead, stick may be cracked. Some predeployment techniques for key distribution such as [5] will lead to network reconfiguration when new nodes join. Towards key management infrastructure, a trusted third party which stores the keys is seldom present in BAN scenarios. Even though a trusted third party exists, it still carries the risk of compromise and associated liability. In some instances, Diffie-Hellman (DH) and its variants [6] such as ECDH (Elliptic Curve Diffie-Hellman) have been used for key exchange. However, based on computational security, they are not cost effective for resource-constrained sensors [7].

In order to overcome the limitations mentioned above, an information-theoretical secure solution has been developed, which is based upon the distinct subjective property from environment that nodes can observe. Since the environment around human body is dynamic and complex, physiological signals are quite unique at a given time. Therefore, the idea using physiological signals for securing intersensor communications was first introduced by Cherukuri et al. [8]. Motivated by this initial idea, [9–12] exploited interpulse interval (IPI) to generate cryptographic keys. Though physiological signals measured from different areas of the body have high correlation, these signals do not possess the exact same values [3]. Consequently, when symbols in the keys get reordered, different values could be produced by translational and rotational errors [13]. To avoid the errors, [3] used physiological signals to facilitate key agreement instead of generating keys. Nevertheless, nodes in the same BAN need to measure the same physiological signals, which unavoidably lead to an additional hardware cost and restrained deployed positions (ECG needs to be measured near cardiovascular).

The above difficulties are real challenges faced by researchers attempting to find another subjective property. Recent works such as [14, 15] have indicated that wireless channel could be the next common property to share a secret. In wireless environment, multipath is a basic character, which brings in a rapid decorrelation with distance. Therefore, the wireless channel between two nodes, Alice and Bob, can produce a special mapping between transmitted and received signals for a shared secret key agreement. Based on the dependence from position and motion, the channel can not be inferred by an eavesdropper, Eve, who is more than half of wavelength away from both Alice and Bob. Obviously, BAN could easily satisfy the condition of location and movement because of human activity. Considering resource-constrained condition, BAN requires periodic reasonably key renewal with a minimum hardware and software overhead. Exiting channel property based key-sharing solutions are established to extract secret keys at a high rate with high bit mismatch rate or generate secret keys at a low rate with acceptable bit mismatch rate. The former needs reconciliation phase to exchange the mismatch information, which consumes resources. The latter is not time effective. It is challenging to optimize two metrics at the same time and balance the trade-off between them.

In this paper, we propose wavelet-transform trend-based key extraction (WTKE) scheme and fuzzy vault based key distribution (FVKD) protocol, which use the received signal strength (RSS) variations for a pairwise key agreement to secure BAN communications. WTKE exploits wavelet transform for more accurate trend to achieve a lower bit mismatch rate than previous work, and it also has an acceptable secret key rate. FVKD, which combines channel property features and fuzzy vault [13] together, could distribute a secret key derived by one end of communication link to the other end. Through experiments, we validate its cost is less than that of DH and ECDH.

The rest of the paper is organized as follows. Section 2 briefly introduces some fundamentals of our paper. Section 3 is devoted to the basic aspects of our key generation scheme WTKE, experimental results, and analysis. Section 4 elaborates on fuzzy vault based key distribution protocol FVKD and further analyses security and performance. Section 5 generally introduces the related work. In Section 6, we come to the conclusion and discuss some possible future directions.

2. Preliminary

In this section, we will introduce channel models in BAN, the fundamentals of key agreement based on channel property, wavelet transform, and fuzzy vault.

2.1. Channel Model in Body Area Network. As sensors' positions are on or inside the body, the BAN channel models need to consider the impact of human body and activities. In [16], IEEE 802.15.6 group defines 3 types of nodes as follows: (1) implant node: a node that is placed inside the human body; (2) Body surface node: a node that is placed on the surface of the human skin or at most 2 centimeters away; (3) external node: a node that is not in contact with human skin. Based on the different node types, [16] gives 4 channel models (CM) shown in Table 1. In this paper, with the hardware constraints, we focus on the latter two channel models (on-body and off-body).

2.2. Key Generation Based on Channel Property. There are three reasons why channel is regarded as secret information.

(1) Reciprocity of electromagnetic propagation: as wireless channel is symmetric, the multipath characters of the radio channel (gains, phase shifts, polarisation distortions, and delays), which Alice and Bob can get from the received signal, are identical.

(2) Temporal variations in the radio channel: The motion by either Alice, Bob, or other objects in the environment near the link would make the channel change over time. Evidently, the security mechanism can reap the full benefit of the randomness resulting from channel variations.

(3) Spatial variations: the radio channel depends on the location of Alice and Bob. Only if another node Eve is more than half a wavelength away from both of them, she will get a different channel.

Therefore, we can take the wireless channel as a time- and space-varying filter. The combination of channel properties (gains, phase shifts, polarisation distortions, and delays) can represent different channels uniquely. Among those channel properties, received signal strength (RSS), which can be

Table 1: Channel models.

Description	Channel model
Implant to implant	CM1 (in-body)
Implant to body surface or external	CM2 (in-body)
Body surface to body surface	CM3 (on-body)
Body surface to external	CM4 (off-body)

measured by most of the off-the-shelf wireless cards, is widely used. According to [17], the on-body BAN wireless channel exhibits reciprocity. From [15], we can infer that the off-body BAN wireless channel will reveal reciprocity if mobility is brought in.

When Alice and Bob are sampling the channel with Eve locating more than half a wavelength away, the valid samples are limited by the rate of time variation. And this variation can be approximately analyzed by using the level-crossing rate (LCR): for a Rayleigh fading process, LCR $= \sqrt{2\pi} f_d \rho e^{-\rho^2}$ [18], where f_d is the maximum Doppler frequency and ρ is the threshold level. Setting $\rho = 1$ gives LCR $\sim f_d$. Thus, the time variation can be quantified by f_d, which could be calculated as

$$f_d = \frac{v}{\lambda}. \tag{1}$$

In (1), v is a measure of the effects of dynamic change by sensors and environment, but, as opposed to other dynamic changes, sensors' motion mostly plays a dominant role in BAN. Thus, we use relative movement velocity between two nodes to signify dynamic change here. And λ is the wavelength of the carrier wave, which can be easily gotten from (2):

$$\lambda = \frac{c}{f_0}, \tag{2}$$

where c is the speed of light. Taking 2.4 GHz frequency and $v = 1\,m/s$ as an example, $\lambda = 0.125$ m and $f_d = 8$ Hz, implying that the maximum useful probes are 8 in a second and Eve should be at least 6.25 cm away from Alice and Bob. As a result, if Alice is sender, Bob should send ACK back in channel coherence time period (a rough estimate is $1/f_d = 125$ ms) after received Alice's signal. Even if Alice and Bob sample the channel at a higher rate, it would be a waste because of many consecutive duplicated RSS values. These duplicate RSS values do not make a contribution to shared information between Alice and Bob. In [19], channel coherence time is defined as the period when the autocorrelation coefficient is above 0.7. In [20], temporal on-body channel coherence time is calculated. According to [21], the longer Bob's response delays during channel coherence time period, the lower correlation of the RSS values measured by Alice and Bob will be. Furthermore, based on [15], channel variation and the reciprocity between Alice's and Bob's RSS values are related. The correlation is the foundation for Alice and Bob to get a common secret key, which can be quantified by using the Pearson correlation coefficient r

$$r = \frac{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)^2}\sqrt{\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2}}, \tag{3}$$

where X_i and Y_i are the RSS values of the ith packet of Alice and Bob and $\overline{X}(\overline{Y})$ is the mean RSS values of Alice (Bob). When r is equal to 1, it means Alice's RSS values match Bob's perfectly. But there is no chance for ideal situation to be achieved in practice for reasons including random noise, asymmetrical interference, or transceiver differences. The offset often leads to mismatch in key generation. Traditional approaches to solve this problem are information reconciliation and exploiting a lossy quantization like [14, 21–23]. However, due to the channel reciprocity, Alice's and Bob's measurements should have similar fading trend, which is analyzed for key extraction through *wavelet transform*, and it turns out that the probability of mismatch decreases but still exists. To address this drawback, a fuzzy method called *fuzzy vault* is used in this paper.

2.3. Wavelet Transform. Wavelet transform is the representation of a function by wavelets, which is mathematical functions used to divide a given function or continuous-time signal into different scale components. Wavelet transforms are classified into discrete wavelet transform (DWT) and continuous wavelet transform (CWT). Take CWT to illustrate the definition: for a given signal $f(t) \in L^2(R)$, its CWT is

$$WT_{\psi}\{f\}(a,b) = \langle f, \psi_{a,b}\rangle = \int_R f(t)\,\psi_{a,b}(t)\,dt, \tag{4}$$

where

$$\psi_{a,b}(t) = |a|^{-1/2}\psi\left(\frac{t-b}{a}\right). \tag{5}$$

ψ of the above formula is called a mother wavelet, and it is also in $L^2(R)$, while a is positive and defines the scale and b is any real number and defines the shift.

The wavelet transform is often compared with the Fourier transform, in which signals are represented as a sum of sinusoids. The main difference is that wavelets are localized in both time and frequency whereas the standard Fourier transform is only localized in frequency. In other words, wavelet transforms have advantages over traditional Fourier transforms for representing functions that have discontinuities and sharp peaks, and for accurately deconstructing and reconstructing finite, nonperiodic, and/or nonstationary signals. And the discrete wavelet transform is also less computationally complex, taking $O(n)$ time as compared to $O(n \log n)$ for the fast Fourier transform. Since the channel changes randomly, our RSS measurements compose an aperiodic signal. Obviously, wavelet transform is better than Fourier transform in our case.

2.4. Fuzzy Vault. The fuzzy vault was first presented by [13] as a cryptographic construction where one can hide a secret (S) in a vault by using the set A. Others can get the secret only when the vault is unlocked on the condition that their set B is closed to set A (B has enough number of values in common with A). The procedure of locking is as follows. (1) Create a v degree polynomial p over the variable x; (2) Use set A as the values of x to compute the values of p and create a set

```
**Public Parameters**: field $F$
**Input**: set $A = \{a_i\}_{i=1}^{t}$, secret $S$, chaff number $r$
**Output**: vault $R$
$R \leftarrow \emptyset$;
$p \leftarrow S$;
$i = 1$;
**while** $i \le t$ **do**
    $(x_i, y_i) \leftarrow (a_i, p(a_i))$;
    $R \leftarrow R \cup (x_i, y_i)$;
**end while**
**while** $t + 1 \le i \le r$ **do**
    $x_i \in F - A$;
    $y_i \in F - \{p(a_i)\}$;
    $R \leftarrow R \cup (x_i, y_i)$;
**end while**
```

Algorithm 1: Fuzzy vault lock.

$R = \{(a_i, p(a_i))\}$, where $a_i \in A, 1 \le i \le |A|$; (3) randomly generate a set $C = \{(c_i, d_i)\}$, where $d_i \ne p(c_i)$, and then add C to R. It is depicted in Algorithm 1, where $p \leftarrow S$ means hide the secret S into polynomial p (e.g., taking S to be the coefficients of p).

To unlock the above vault, others use their set B to recall the specified point of set A in R. If B gets more than $v+1$ right points, the right polynomial p can be reconstructed to obtain the secret S. Process is showed in Algorithm 2.

FVKD can map this scheme through setting the channel property obtained at the sender to set A, those obtained at the receiver to set B, and the secret key that needs to be shared to polynomial coefficients.

3. Wavelet-Transform Trend-Based Key Extraction

In this section, we will propose basic idea of wavelet-transform trend-based key extraction, analyze the security of the scheme, and evaluate the performance of the scheme using real collected data.

3.1. Scheme Description. As shown in Figure 1, the complete process is divided into the following steps.

3.1.1. Sampling. First, both Alice and Bob should obtain common information based on RSS value. Alice sends a probe message periodically for Bob to get RSS values, and after Bob receives message, he sends back probe message as soon as possible for Alice to sample the channel. As a result, Alice and Bob get related almost-exact values, respectively. The sample process lasts for a fixed duration at a specific rate to get enough samples, and all of samples must be obtained on the condition that channel is changing to keep the reciprocity as described in Section 2.

```
**Public Parameters**: field $F$
**Input**: set $B = \{b_i\}_{i=1}^{t}$, vault $R = \{(x_i, y_i)\}_{i=1}^{r}$
**Output**: set $Q$
$Q \leftarrow \emptyset$;
$i = 1$;
**while** $i \le t$ **do**
    $(x_i, y_i) \leftarrow$ search $R$ at $(b_i, 0)$;
    $Q \leftarrow Q \cup (x_i, y_i)$;
**end while**
```

Algorithm 2: Fuzzy vault unlock.

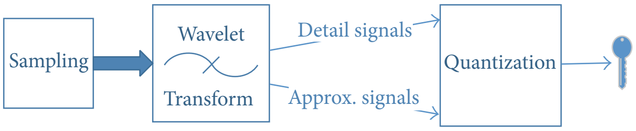


Figure 1: Flow chart of key extraction.

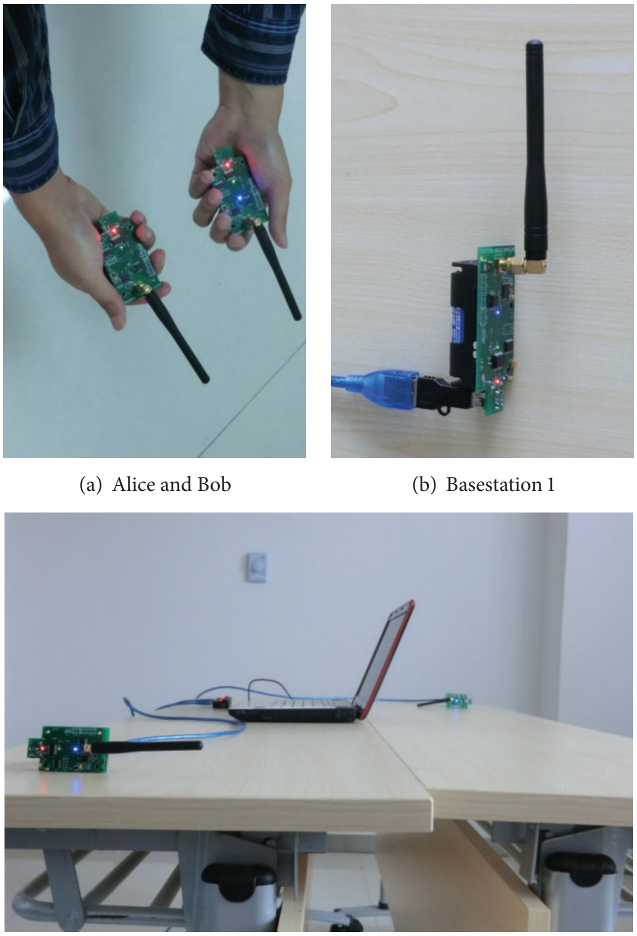
3.1.2. Wavelet Analysis. As a number of factors such as random noise, asymmetrical interference, transceiver differences, and half-duplex communication bring in some mismatches between the original RSS readings of both sides, but they have the same fading trend. In order to accurately evaluate the trend, we use wavelet transform. Through wavelet transform, signal is decomposed into multiresolution levels; thus the corresponding detail and approximation signals are obtained. At the same level, approximation signal represents the low frequency part, and detail signal shows the high frequency part. Clearly, we can use approximation signal to get the same trend. There are many mother wavelet functions for different purposes or signals. Here we use Haar wavelet because of its advantage for the analysis of signals with sudden transitions. In addition, we choose 4-level wavelet transform, and the reasons are given in the following experiments and performance evaluation.

3.1.3. Quantization. Through Haar wavelet transform, signal will have sudden changes. We can therefore detect those changes and make binary quantization. For an RSS measurement $r(k)$, we detect sudden transitions through function as follows:

$$\Phi = r(k+1) - r(k). \tag{6}$$

If $\Phi > 0$, which means the wave rises, we extract a 1 from the measurements. When $\Phi < 0$, which means the wave declines, a 0 is generated.

3.2. Experiments. We conduct our experiments in typical indoor environments. Our experiments were divided into two parts. In scenario 1, we verify our scheme through on-body channel by using two sensors in the same BAN. In this situation, sensors provide real time service, so sensors need to probe the channel frequently. In scenario 2, we exploit sensor-to-basestation communications to test our scheme on off-body channel, and sensors on the body keep a low-data-rate communication with basestation. For this reason, we can use the scheduled communication to measure RSS value.



(a) Alice and Bob

(b) Basestation 1

(c) Basestation 1 and Basestation 2

Figure 2: Mobile node and basestations.

In the existing hardware system, the RSS values are reported as integers through wireless card, but different products from different companies calculate RSS values with deviations. “For example, Atheros devices report RSS values from −35 dBm to −95 dBm, Symbol de-vices report RSS values from −50 dBm to −100 dBm, in 10 dB steps, and Cisco devices report RSS values in the range −10 dBm to −113 dBm [15].” To avoid the impact of different manufactures, we use products from the same corporation.

Our experiments used Telosb nodes running TinyOS and operating in the 2.4 GHz band. With logarithmic units, their radio antennas (CC2420) can measure signal power and output a received signal strength indicator (RSSI). Our setup is listed in Table 2. For scenario 1, Alice and Bob are mounted on the body (here we take them in hand), shown in Figure 2(a). And Alice transmits a probe packet every 40 ms, and Bob sends message back once he receives the packet. For scenario 2, Alice is still a body-worn node, but the probe interval is changed to 80 milliseconds. The basestation 1 (Bob, shown in Figure 2(b)) still replies an acknowledgement immediately after he receives the packet. The settings above allow the two endpoints of the communication link to sample the channel alternately in quick succession to ensure the high reciprocity between their measurements.

Table 2: Experimental setup.

	Scenario 1	Scenario 2
Sensor in the left hand	Alice	Alice
Sensor in the right hand	Bob	—
Basestation 1	—	Bob
Basestation 2	Eve	Eve
Channel description	On-body	Off-body
Probe interval	40 ms	80 ms

The layout of our indoor environment experiments is depicted in Figure 3 showing the location of the basestations (as shown in Figure 2(c)). The room is covered by multiple WiFi networks, which may result in interference. The subject walked around the table to ensure that the channel between two parties of the link varies. In scenario 1, the on-body channel varies with arm swinging. In scenario 2, it is obvious that movement of the subject can bring in changes to the off-body channel between Alice and Bob (basestation 1). In

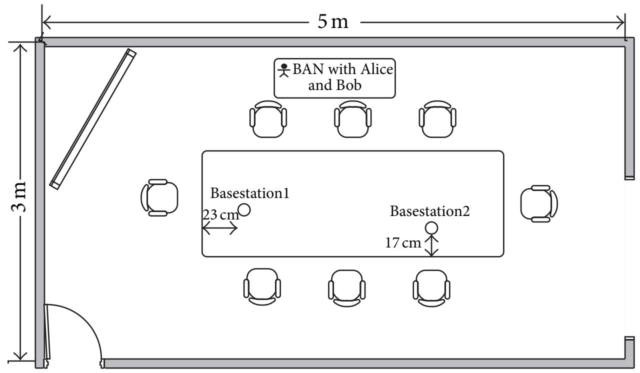


Figure 3: Experimental layout for indoor environment.

both experiments, the subject walked and waved just as we normally do in real life.

For scenario 1, we show the signal strengths measured by Alice, Bob, and Eve in Figure 4. It can be obviously observed that the eavesdroppers are not able accurately to replicate the channel measurements between Alice and Bob, and it can also be noticed that Alice's and Bob's RSS values have a similar trend. As a result, the RSS measurements are the distinct information shared between Alice and Bob to generate secret key or channel features. However, we find that even though Alice' RSS values are close to Bob's, their values are not exactly the same and discrepancies may result in mismatches of secret key extraction (our channel feature generation). The discrepancies between channel properties measurements are similar to deviation between the physiological signal measurements, which reminds us of using trend- and fuzzy-based methods. The result of experiment in scenario 2 is shown in Figure 5, which seems similar to that of scenario 1. As the channel between Bob (basestation 1) and Eve (basestation 2) is almost steady, the RSS trace of Eve from Bob is not painted.

Figure 6 shows the autocorrelation coefficients for Alice and Bob in two scenarios. As the definition is mentioned in Section 2, the channel coherence time is about 140 ms in scenario 1 (on-body channel) and 50 ms in scenario 2 (off-body channel) based on the observations of Figures 6(a) and 6(b). We denote the link Alice to Bob as h_{ab} and the link Bob to Alice as h_{ba}. Figures 7 and 8 illustrate the scatter plots and the histograms of link power difference $h_{ba} - h_{ab}$ with normal density in two scenarios, respectively. Exact reciprocity occurs when $h_{ba} = h_{ab}$, and the difference between Alice's and Bob's measurements is small, which validates the reciprocity of the on-body and off-body channels. Parts of approximation signals in scenario 1 are exhibited in Figure 9. Obviously, after 4-level wavelet transform, most changes have been removed. It will make Eve's trend more similar to legitimate devices. That is one of the reason why we choose 4-level wavelet transform.

3.3. Security Analysis. For our threat model, our first assumption is the adversary Eve own unlimited computation capability. Eve can eavesdrop all the traffic between Alice and Bob and can also sample the channel at the same time when Alice and Bob measure the channel, but she only gets information about the channels between herself and either Alice or Bob. Another assumption is that Eve knows the algorithm and settings. However, Eve cannot be less than half of wavelength close to either Alice or Bob. This assumption is reasonable in BAN as human can easily detect the illegal node a few centimeters away. We do not concern the issue about authentication, for it is another distinct problem, and it should be done in the start-up phase. There are many solutions such as [24–26] to authentication. Those solutions can be used in conjunction with our scheme. Based on the measurements of our experiments, all devices generate keys by using WTKE, and the results of bit mismatch rate are shown in Figure 10. The bit mismatch rate between Eve and legitimate devices is nearly 0.5, much higher than that between legitimate devices themselves. Obviously, our key extraction scheme, fading trend-based wavelet key extraction, has the high security to generate secret key through channel properties measurements.

3.4. Performance Evaluation. It is important to ensure the randomness of a secret key. Therefore, we test our keys generated from two scenarios in the NIST test suite [27]. Because of limitation of the bit length, we only run some tests, and the results are listed in Table 3. To pass a test, the value for that test must be greater than 0.01, and our results meet the requirement.

We compare our method with typical amplitude quantization scheme [14] and previous trend quantization work [28] through three performance metrics: (1) secret bit rate: the average number of secret bits extracted per second; (2) bit mismatch rate: the ratio of the number of bits that do not match between Alice and Bob (3) entropy: The entropy of a random variable X is defined as

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i), \tag{7}$$

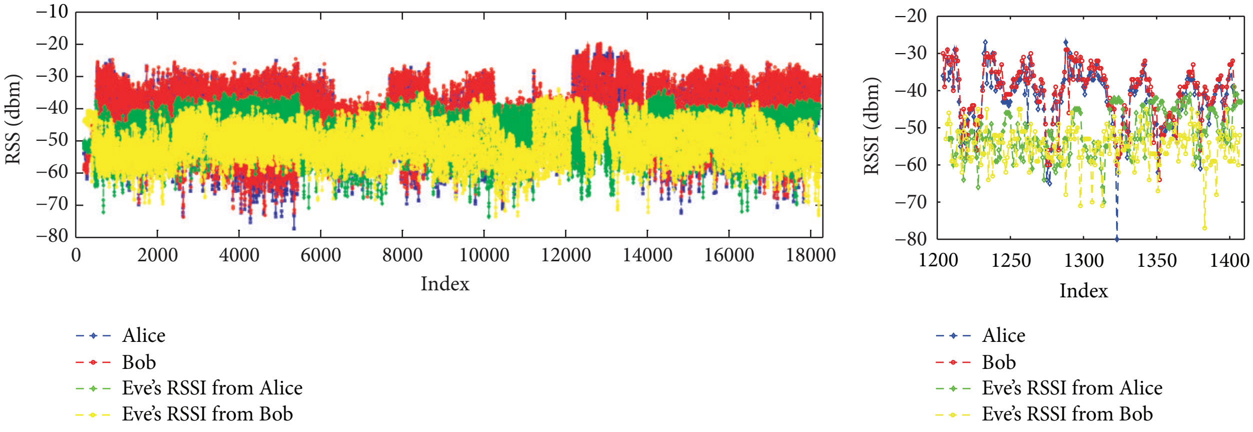
where X is a set of n symbols $\{x_1, \ldots, x_n\}$ and $p(x_i)$ signifies the probability of the symbol x_i [29].

In [14], Mathur et al. construct a quantizer, which calculates two thresholds $q+$ and $q-$ through

$$q+ = \mu + \alpha * \sigma, \qquad q- = \mu - \alpha * \sigma. \tag{8}$$

Here, μ and σ are mean and standard deviation, and α is a parameter which can be configured dynamically on the condition of $0 < \alpha < 1$. And the samples are dropped, if their values are between $q+$ and $q-$. Furthermore, a single 1 or 0 is produced when m consecutive samples' values are greater than $q+$ or less than $q-$. We choose $\alpha = 0.2$ and $m = 5$ in this paper.

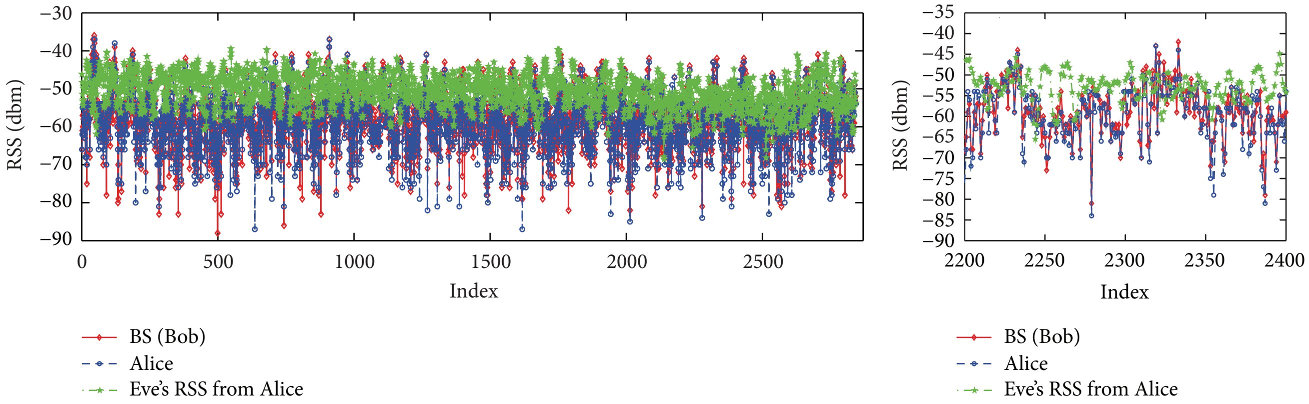
In [28], Liu et al.'s scheme contains both trend and amplitude multilevel quantization. Since WTKE is a single bit extraction scheme, we only take fading trend estimation from Liu et al.'s scheme into account. Fading trend estimation runs as follows For an RSS measurement $r(k)$, it defines $\Phi^1 = r(k) - r(k-1)$ and $\Phi^2 = r(k+2) - r(k)$. If Φ^1 and Φ^2 are



(a) RSS traces in scenario 1

(b) A magnified portion

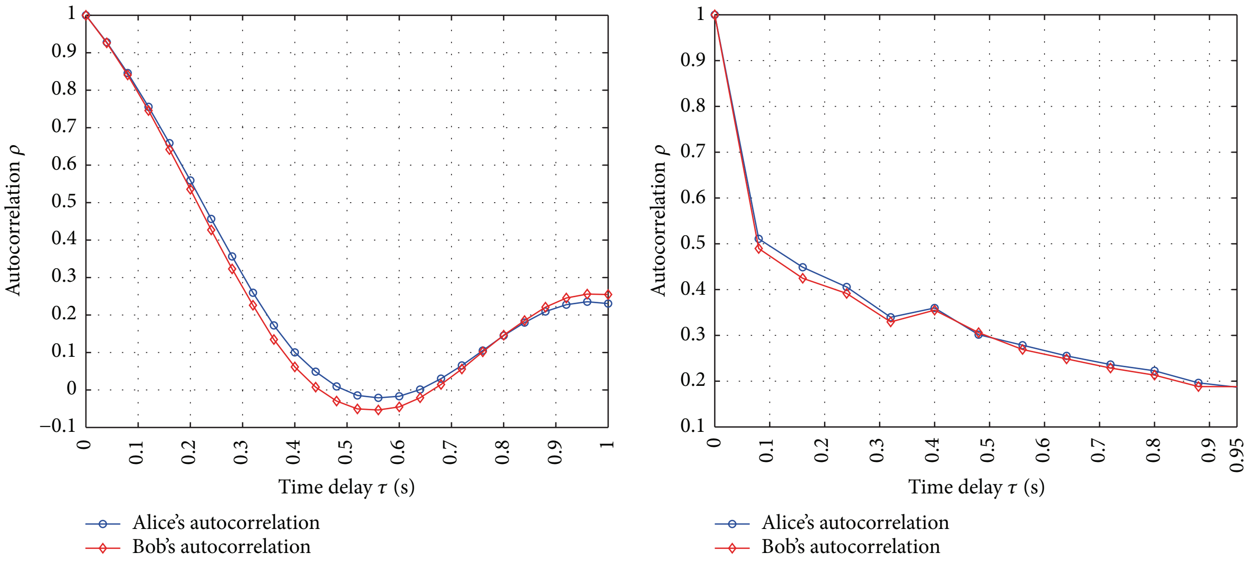
FIGURE 4: Experiments results in scenario 1.



(a) RSS traces in scenario 2

(b) A magnified portion

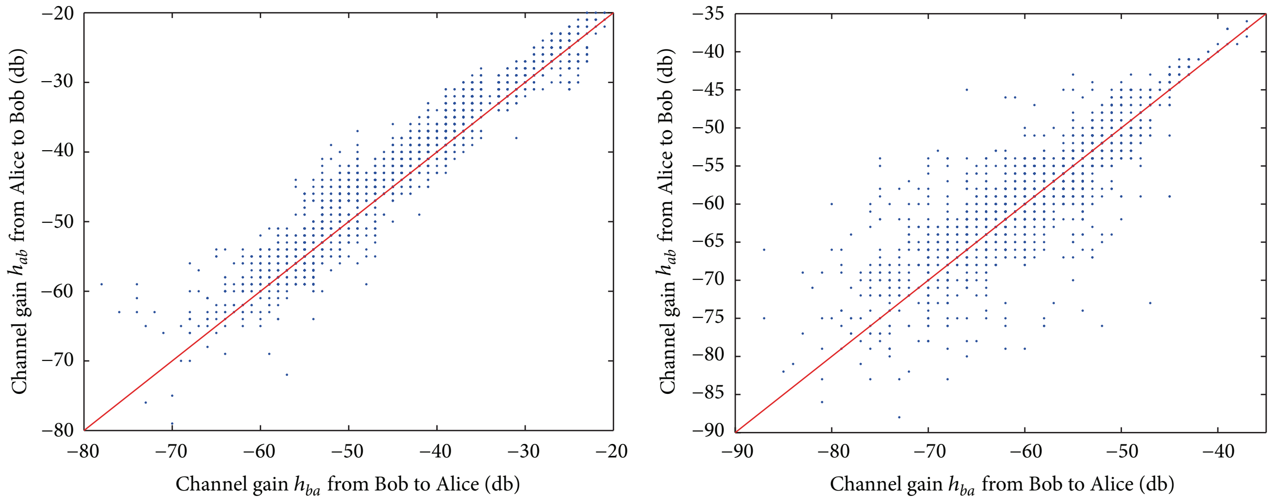
FIGURE 5: Experiments results in scenario 2.



(a) Scenario 1

(b) Scenario 2

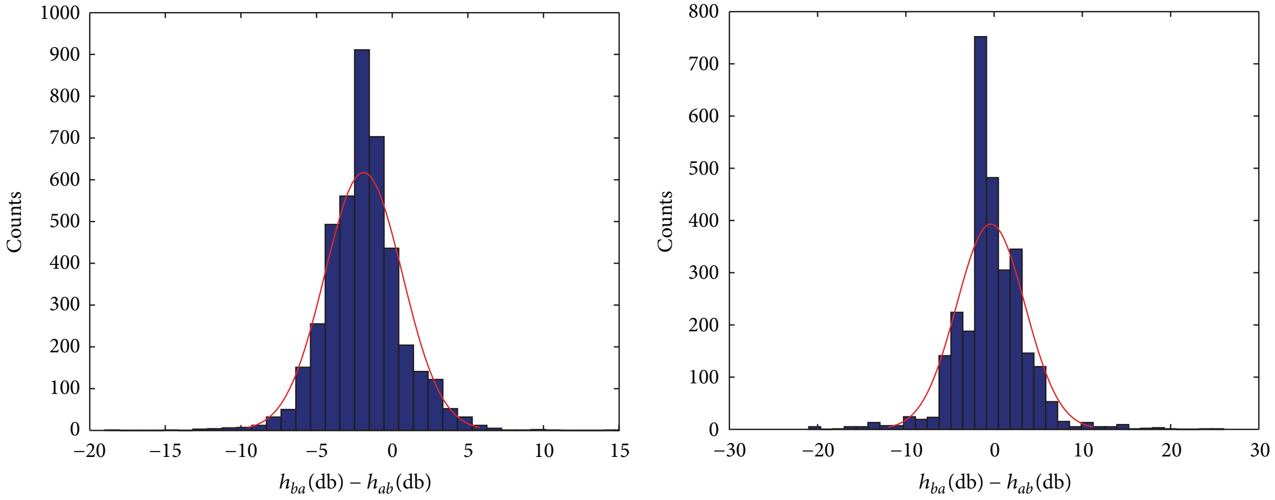
FIGURE 6: Autocorrelation for Alice and Bob in two scenarios.



(a) Scenario 1

(b) Scenario 2

Figure 7: Channel gain scatter plot h_{ba} versus h_{ab} in two scenarios.



(a) Scenario 1

(b) Scenario 2

Figure 8: Histogram of channel gain h_{ba} versus h_{ab} with normal density in two scenarios.

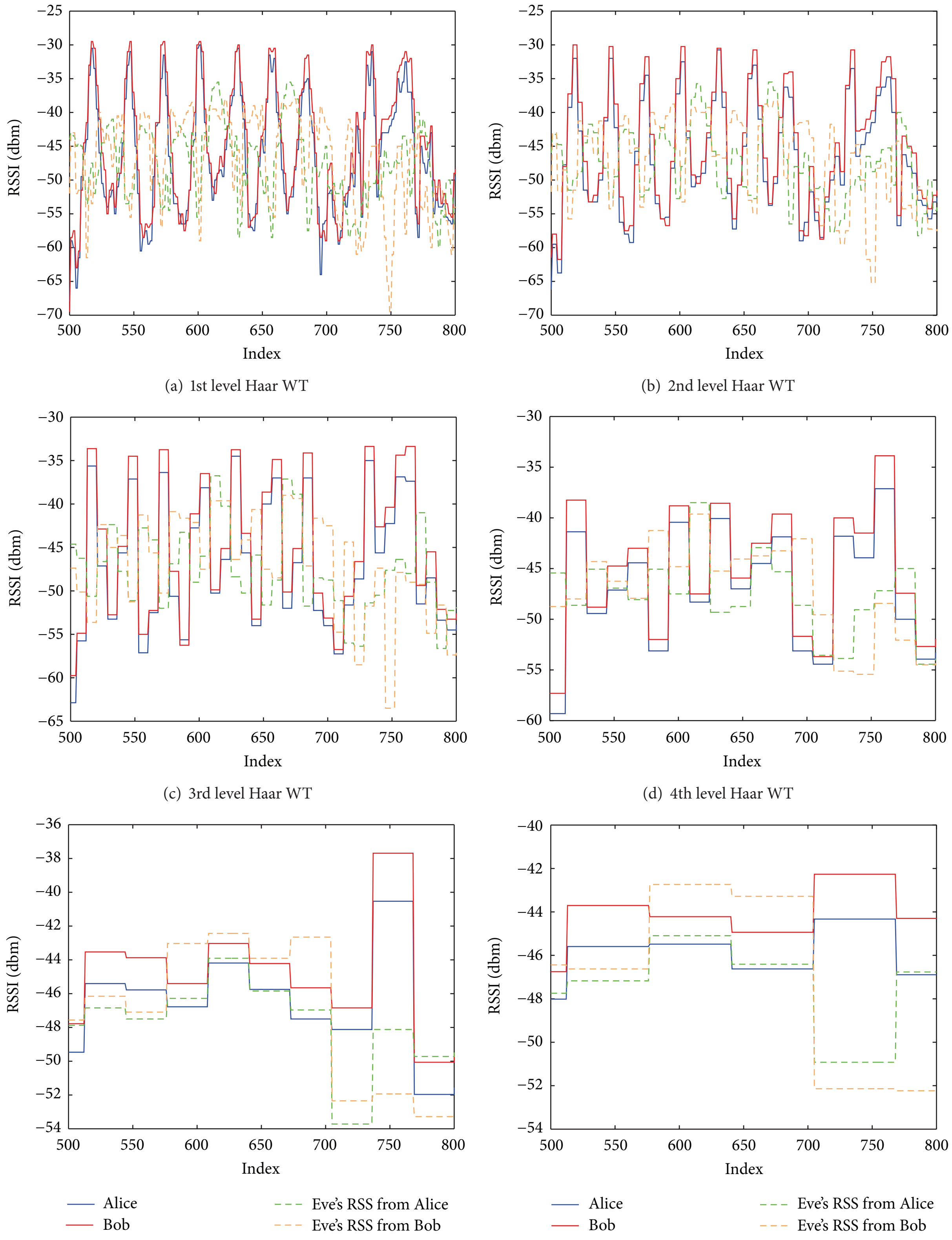
Table 3: NIST statistical test suite results.

Test	Scenario 1	Scenario 2
Frequency	0.25	0.34
Block frequency	0.04	0.24
Cumulative sums (Fwd)	0.17	0.03
Cumulative sums (Rev)	0.16	0.19
Runs	0.97	0.19
Longest run	0.44	0.22
Approx. entropy	0.87	0.83
Serial	0.74 0.41	0.74 0.41

positive or negative at the same time, a single secret bit could be extracted.

The performances of all three schemes in two scenarios are exhibited in Figures 11, 12, and 13. Though Liu et al.'s scheme has the highest secret bit rate, they get the highest bit mismatch rate. Our WTKE scheme is better than Mathur et al.'s scheme in both aspects of secret bit rate and bit mismatch. All the schemes produce bit streams with nearly the same high entropy. In a word, both trend-based methods are better than Mathur's amplitude-based methods. Compared to Liu's scheme, our method accurately gets the trend through wavelet analysis, which also reduces the frequency of wave change.

Here, we present another reason why we choose 4-level Haar wavelet transform. As shown in Figure 14, the curve of secret bit rate goes down with the curve of bit mismatch. The curve of bit mismatch falls faster before 5 Level, and the curve of secret bit rate falls faster after 5 Level. In a word, it is



(a) 1st level Haar WT

(b) 2nd level Haar WT

(c) 3rd level Haar WT

(d) 4th level Haar WT

(e) 5th level Haar WT

(f) 6th level Haar WT

Figure 9: Parts of approximation signals from Haar wavelet transform in scenario 1.

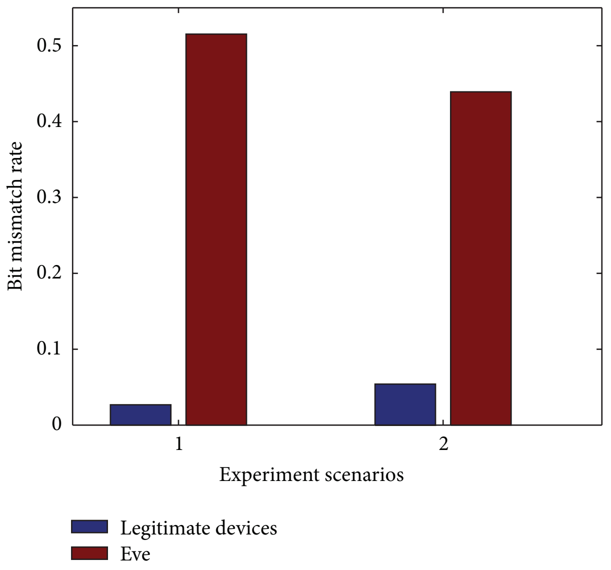


Figure 10: Bit mismatch rate for legitimate devices and Eve under different scenarios.

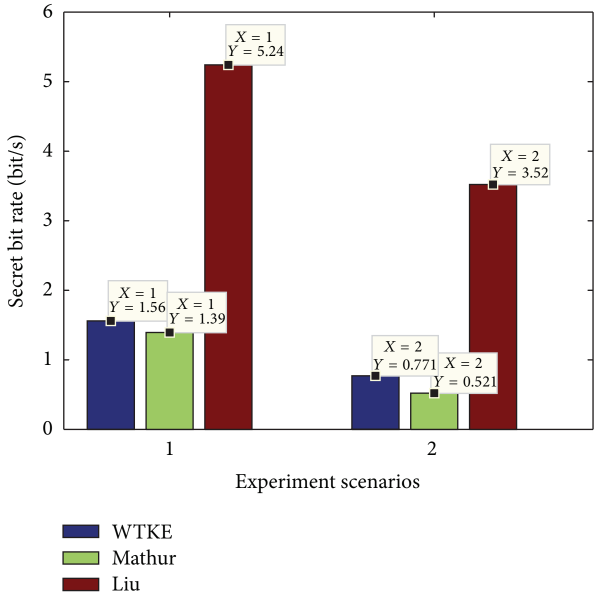


Figure 11: Secret bit rate comparison.

worth sacrificing secret bit rate to reduce mismatch before 5 level.

4. Fuzzy Vault Based Key Distribution

In this section, we first describe our novel key distribution protocol based on fuzzy vault. Then, we give analysis on its security and efficiency.

4.1. Protocol Description. Our key distribution protocol uses fuzzy vault to facilitate a pairwise symmetric key agreement between Alice and Bob through measuring RSS values. The protocol is as follows.

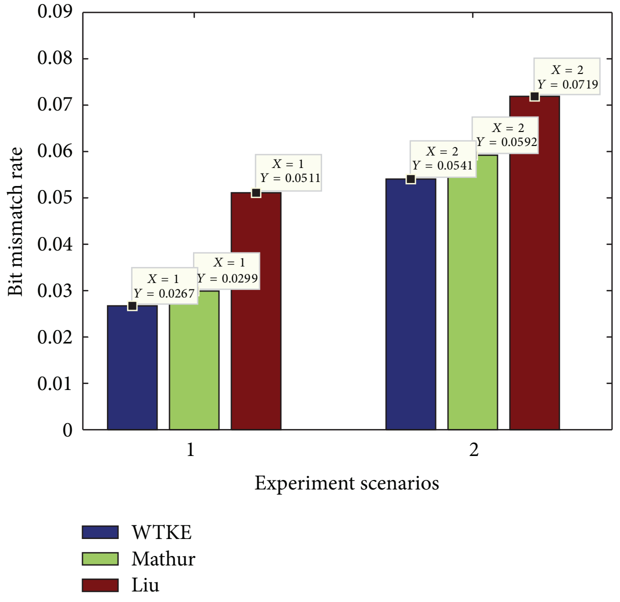


Figure 12: Bit mismatch rate comparison.

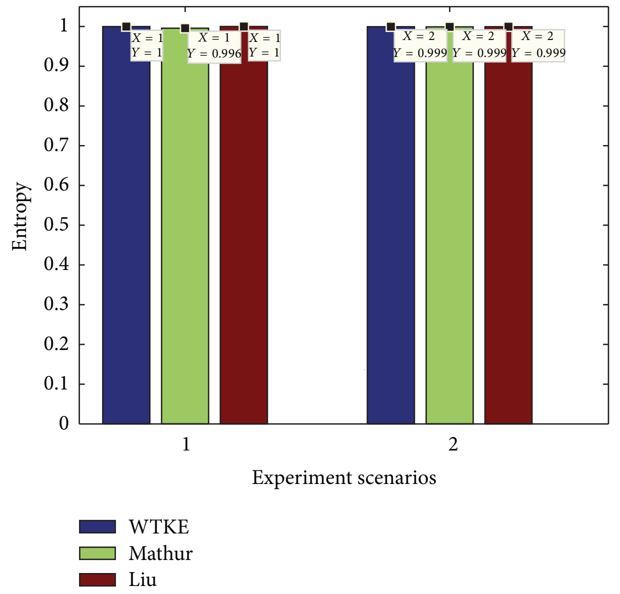


Figure 13: Entropy comparison.

4.1.1. Channel Feature Generation. First, both Alice and Bob should obtain common information based on RSS value. Next, every 128 samples form a window, and neighboring windows have 32 overlapping samples. Using WTKE mentioned in Section 3 to process each window, we will get 8-bit binary. In order to increase the range of chaff points, 4-level Daubechies 5 wavelet transform is used to search for the fourth level detail signal's maximum point, whose position in 128 samples will be converted into 7-bit binary. For example, as shown in Figure 15, if the window contains 128 samples from index 500 to 627, the maximum points of Alice and Bob in 128 samples are both at index 518. So the position of

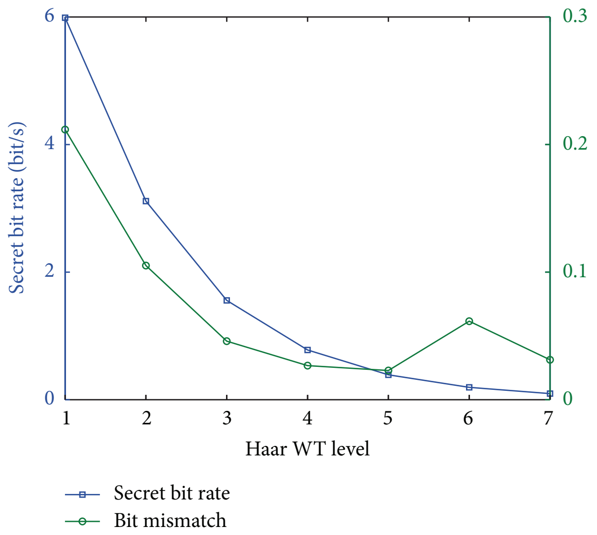


FIGURE 14: Secret key rate versus bit mismatch rate in scenario 2.

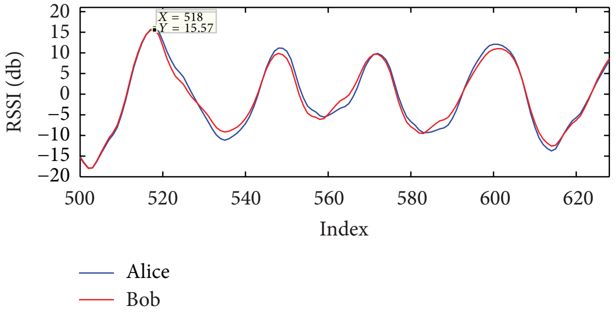


FIGURE 15: 4th level db5 detail signal.

maximum points in the window is 19 and could be converted into 0010011. Finally, the first 6 bits generated by WTKE and all 7 bits extracted by maximum point form a channel feature. A group of channel features generates a feature vector $F_D = \{f_D^1, f_D^2, \dots, f_D^N\}$, where D is either the sender Alice or receiver Bob and N is the size of the feature vector. In the end of the feature generation process, Alice owns $F_A = f_A^1, f_A^2, \dots, f_A^N$, and Bob has $F_B = f_B^1, f_B^2, \dots, f_B^N$. The whole flow chart is shown in Figure 16.

4.1.2. Polynomial Choice. Alice uses a random generator to produce random numbers which is the secret key she wants to share with Bob. The length of the key is set as 128 bits here, and it could also easily be set longer. To create a vth order polynomial of the form $p(x) = c_v x^v + c_{v-1} x^{v-1} + \cdots + c_0$, the key is divided into $v+1$ parts to form the coefficients $\{c_i\}_{i=0}^{v}$ of the polynomial. The order v of $p(x)$ is not a secret and is known to all sensors in the BAN. If Bob reconstructs the polynomial successfully, he can get the secret key by concatenating the coefficients together (key $= c_v | c_{v-1} | \cdots | c_0$).

4.1.3. Vault Creation. Since the polynomial and feature vectors are now available, fuzzy vault is then created by computing the set $P = \{f_A^i, p(f_A^i)\}$, where $f_A^i \in F_A$ and $1 \le i \le N$. To hide the secret points, a set C of M random chaff points also needs to be calculated: $C = \{cf_j, d_j\}$, where $cf_j \notin F_A$, $d_j \ne p(cf_j)$, and $1 \le j \le M$. The value cf_j fluctuates within the same limits as that of the features (2^n, and n is the length of channel feature). Therefore, the total number of points in the vault ($|R|$) is bounded within 2^n, which is equal to $|M| + |N|$. We define $|R|$ as *vault size*.

4.1.4. Vault Locking. Alice builds the vault $R = P \cup C$ and randomly permutes the values, to make sure the chaff points and the legitimate points are indistinguishable. According to [3], different numbers of points in set C mean different levels of security. With the number of points in set C increasing, the level of security rises. We can set the level of security by changing the cardinality of set C when required.

4.1.5. Vault Exchange. Alice then sends the vault R to Bob using the following message: Alice → Bob: IDs, Nonce, R, Par, MAC (Key, R|Nonce|IDs). Here IDs is the id of Alice, Nonce is a unique random number for transaction freshness, Par is parity of P to improve efficiency of reconstructing polynomial for Bob, and MAC is the message authentication code. Bob can compute MAC to confirm whether the key calculated from the vault is right or not.

4.1.6. Vault Unlocking. Once Bob receives the vault R, he makes parity check using Par and drops the part which does not match F_A. Then, Bob computes the set Q, where $Q = \{(b, c) | (b, c) \in R, b \in F_B\}$. Based on the points in Q, Bob can try to use the Lagrangian interpolation for reconstructing the polynomial p. According to the suggestion of [30], with the knowledge of $v+1$ points $\{(x_0, y_0), (x_1, y_1), \dots, (x_v, y_v)\}$ on a vth order polynomial, we can rebuild the polynomial by performing the following linear combination: $p'(x) = \sum_{j=0}^{v} y_j d_j(x)$, where $d_j(x) = \prod_{i \ne j, i=0}^{i=v} (x - x_i)/(x_j - x_i)$. The condition $|Q| > v$ must be satisfied for Bob to unlock the vault successfully. Obviously, Bob tries to unlock the vault by taking $v+1$ points from Q every time. To check the validity of the unlocking, Bob verifies the MAC by concatenating the coefficients of the resulting polynomial concatenated together.

4.1.7. Vault Acknowledgment. If unlocking is successful, Bob sends a message back to Alice to inform the successful unlocking. The message is described as follows: Bob → Alice: MAC (Key, Nonce|IDs|IDr). The meanings of Key, Nonce, and IDs are described in the paragraph of Vault Exchange, and IDr means the ID of Bob. After verifying the acknowledgement sent by Bob, Alice knows the key has been shared successfully. Since Bob measures the similar distinct RSS values to the measurements of Alice, only Bob can unlock the vault.

Figure 17 shows the key distribution process. This protocol provides a one-hop security by channel property, and this protocol can be easily extended to multihop end-to-end communication, where the estimates of shared channel

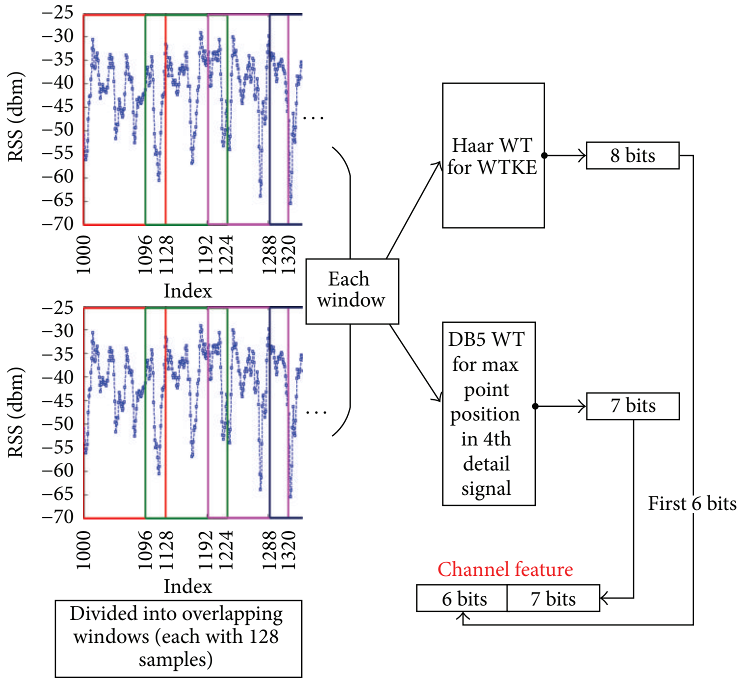


Figure 16: Flow chart of channel feature generation.

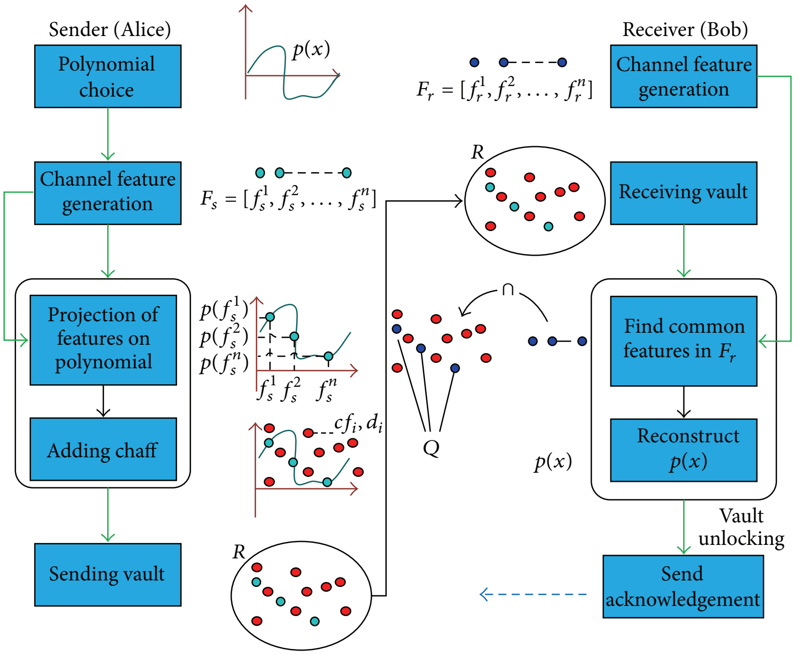


Figure 17: Fuzzy vault based key distribution protocol.

property could be delivered as proposed in [28]. Considering the latency problem in BAN, sensors can initialize a secret key by only executing the protocol one time and then derive keys from this initialized key for ensuring the security of more communications.

4.2. Security Analysis. With the use of fuzzy vault, FVKD ensures that the two ends of the communication link can share a secret key though they do not have all the same features. The security of the FVKD scheme depends on the difficulty of polynomial reconstruction. We can hide the legitimate feature points among a mass of the spurious chaff points, whose values fluctuate in the same range as that of legitimate ones. An adversary, without knowledge of legitimate points, has to try out each of the $v+1$ points in set R to get the correct polynomial, and the probability of success remains very low.

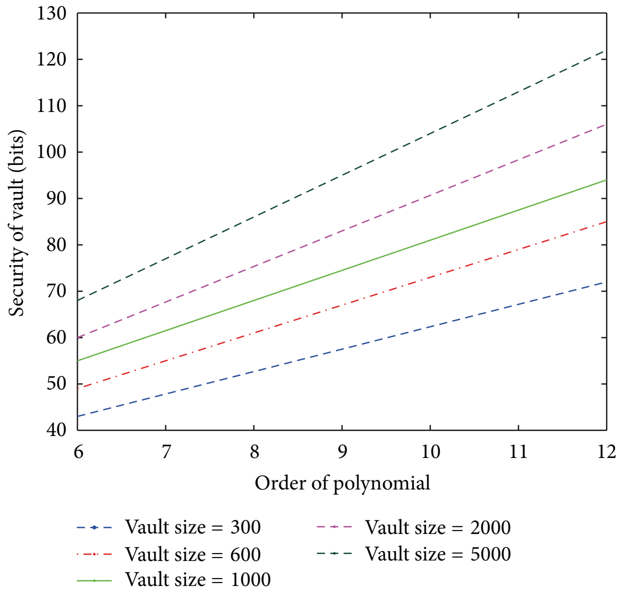


FIGURE 18: Security of the vault.

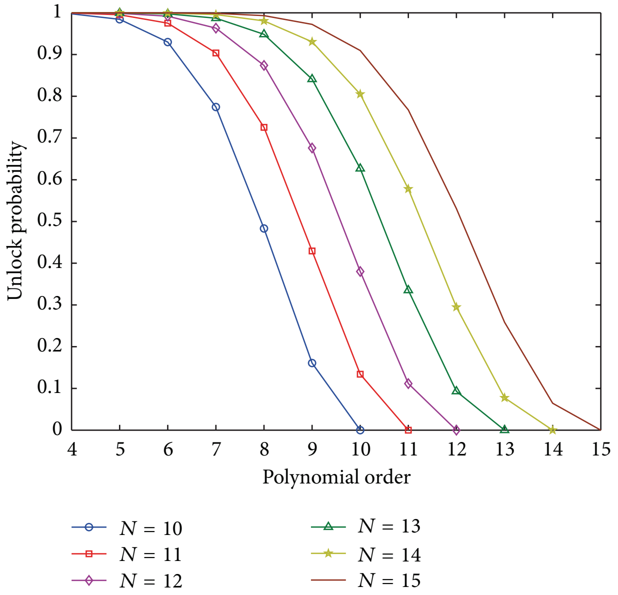


FIGURE 19: Unlock probability.

Obviously, if Bob acquires more features, it gets easier to reconstruct the hidden polynomial. When there are more chaff points or higher order polynomial, the vault becomes more secure. According to [3], as shown in Figure 18, with 2000 chaff points and order, security strength of the vault is the same as 60 bits when polynomial order is 6, and security level is approximately 105 bits when polynomial order is 12.

It is also hard for adversaries to know the key in the vault exchange and acknowledge phases. As mentioned in Section 3, Alice and Bob could authenticate each other by using methods in [24–26]. It is a reasonable assumption that Alice and Bob have been device paired. For this reason, the existence of ID (IDs and IDr) could be used to control access. The Nonce provides the freshness of the protocol, and the MAC is used to prevent man-in-the-middle attack.

4.3. Performance Evaluation

4.3.1. Distinctiveness. The channel properties depend on the location and relative motion of two endpoints in the link. Half wavelength (e.g., 6.25 cm of 2.4 GHz carrier frequency) is close enough for the BAN user to detect adversaries. We also conduct a series of theoretical analysis here. As to WTKE, after 4-level Haar transform, the bit mismatch probability (P_{bm}) can be expressed below, with the assumption that channel state is Gaussian estimation of channel state and successive sampling values are independent.

$$\begin{aligned} P_{bm} &= P\left(\Phi_A > 0 \wedge \Phi_B < 0\right) + P\left(\Phi_A < 0 \wedge \Phi_B > 0\right) \\ &= \left(1 - F\left(\Phi_A = 0\right)\right) F\left(\Phi_B = 0\right) \\ &\quad + F\left(\Phi_A = 0\right)\left(1 - F\left(\Phi_B = 0\right)\right), \end{aligned} \tag{9}$$

where $F()$ is the cumulative distribution function for Gaussian distribution, Φ is defined in (6) (Φ_A means Alice's result and Φ_B means Bob's). Based on our experiments and observations, each feature's mismatch probability is determined by mismatch probability of WTKE. That means when the probability P_{bm} of the individual bit mismatch is known, the probability P_q of having mismatching features can be expressed as

$$P_q = \sum_{i=1}^{(n-7)} C_{(n-7)}^{i} P_{bm}{}^{i} \left(1 - P_{bm}\right)^{n-7-i}. \tag{10}$$

Here C is combinatorial computing. Therefore, the probability that receiver can unlock the vault is shown in Figure 19. With consideration of the time effectiveness, we set N, the size of feature vector, below 15. It means if response time of sensor is between 20 to 40 ms, the sample time is 29.44 to 58.88 s. To balance security and efficiency, we set polynomial order as 7 and the size of feature vector as 11. As shown in Figure 10, Eve's P_{bm} is nearly 0.5, which results in extremely high value of its P_q.

4.3.2. Length and Randomness. The key shared between Alice and Bob is generated by Alice using a random number generator. Thus, the length and randomness of the key can ensure communication security.

4.3.3. Temporal Variance. As mentioned above, the variation of channel is quantified by the maximum Doppler frequency, and the variation follows Gaussian distribution. It can be concluded that even though adversaries know the previous values, they cannot infer the value of channel properties.

4.4. Implementation. We prototype FVKD by using Verilog HDL to evaluate its cost and performance in hardware. The Xilinx ISE software tool is used for emulating a Virtex V platform (http://www.xilinx.com). The metrics used for

Table 4: Computational cost and memory footprint of our prototype implementation.

Entity	Total cycles	Memory footprint
Sender	8,417	23.22 KB
Recvr.	13,396	21.35 KB

the evaluation are CPU clock cycles and memory footprint. Table 4 shows the details.

Though the computational cost is composed of many tasks, only three kinds of them (feature generation, key hiding, and key unhiding) are crucial. Both sender and receiver have feature generation step. Only sender has the task of key hiding, and receiver is just responsible for key unhiding task. As key unhiding needs more computations than key hiding, receiver runs more cycles. We set a 25 MHz clock, so it would only take a few milliseconds to execute our protocol. And the cost could be amortized in the sampling phase. Since other than generating features, the sampling phases can computerize and transmit some chaff points in the meantime. The memory foot print values are primarily determined by chaff points (2000, 13-bit x-values, and 23-bit y-values), features (13-bit values), and polynomial projections (23-bit values). Compared to our protocol, DH takes more than 320,000 cycles and its variant ECDH takes more than 100,000 cycles, though they consume less memory footprint.

5. Related Work

There has been increasing research on utilizing wireless channel properties to extract secret keys. This approach is based on the theory of [14, 31, 32], and it suggests that a secret key agreement is possible for two parties to achieve through using correlated random variables of channel in the presence of an eavesdropper. Till now, several channel properties such as signal phase, time delay, angle of arrival, and received signal strength (RSS) have been proposed, among which phase difference is firstly proposed in [31]. It measures differential phase of two-tone signal to generate secret keys. Further research is done by using phase difference to improve the efficiency of key establishment in [33]. The impulse response of a wireless channel is estimated from WiFi signals in [14]. And in [22], statistics of the angle-of-arrival (AOA) is used on the condition that AP needs a programmable phased array antenna.

RSS is widely used due to the fact that it can be easily collected through most off-the-shelf radio devices. As to RSS-based methods, some works focus on temporal and spatial variations such as [14, 15, 34, 35]. In [36–38], some other factors like multiple antenna diversity and multiple frequencies are also taken into account. In [14], the authors develop a lossy quantization to extract key bits from a statistical Gaussian channel. And they supply theoretical certification and validate this mechanism in indoor environment. In [15], Jana et al. extend this approach to measure RSS over a single channel in various environments using laptops. And a multibit quantization method is also proposed to improve secret bit rate with privacy amplification. In [35], the authors use a transform to decorrelate secret key bits for a very high bit generation rate (40 bits/s).

Although BAN is a newly emerging wireless sensor network, there is still research already done on this topic. In [39], the authors examine the near-body radio channel for key generation based on simulation modeling. Body-worn scenarios such as patient mobility, different placement of devices on the body, and different modes of motion are considered in [40]. To avoid reconciliation phase in key generation, the authors analyze the reasons of measurement asymmetries and then use Savitzky Golay filter to improve signal correlation [21]. In [38], channel hopping is used to solve the problem of little dynamic channel change that exists, which means the channel coherence time period is too large.

Different from the above works, we use waveform trend and feature to promote key agreement by using wavelet transform and fuzzy vault.

6. Conclusion

This paper presents two main research findings: a key extraction scheme and a secure key distribution protocol for BAN. On the basis of the high correlation in RSS measurements between two communication parties, wavelet analysis and fuzzy vault are utilized in this paper. With the assumption of enough motions in BAN, a secret key can be shared between two sensors, and the key can also be shared between a sensor and a basestation, even though an eavesdropper is present. This paper analyzes the security and performance of the scheme and protocol, and it turns out that WTKE provides a secure key with low mismatch rate, high entropy and acceptable secret key rate. FVKD meets the requirements of key agreement (randomness, distinctiveness, and temporal variance).

There are some problems this paper does not fully explore. For instance, the channel coherence time period lasts longer than node's response time. We would take measures to adjust the channel coherence time in our future research.

Acknowledgments

This research was supported by National Basic Research Program of China (973 Program) under Grant no. 2011CB302702 and National 863 Program of China under Grant no. 2011AA010305.

References

[1] "Apple inc.," http://www.patentlyapple.com/patently-apple/2012/06/inductively-powered-ring-shines-in-fujitsus-iphone-app-patent.html.

[2] "Shimmer research," http://www.shimmer-research.com/.

[3] K. K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, "PSKA: usable and secure key agreement scheme for body area networks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 1, pp. 60–68, 2010.

[4] A. Astrin, "Ieee standard for local and metropolitan area networks part 15.6: wireless body area networks: Ieee std 802.15.6-2012," 2012.

[5] S. Zhu, S. Setia, and S. Jajodia, “LEAP+: efficient security mechanisms for large-scale distributed sensor networks,” *ACM Transactions on Sensor Networks*, vol. 2, no. 4, pp. 500–528, 2006.

[6] A. Liu and P. Ning, “TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks,” in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN ’08)*, pp. 245–256, IEEE, April 2008.

[7] E. Blaß and M. Zitterbart, “Efficient implementation of elliptic curve cryptography for wireless sensor networks,” Tech. Rep., Institute of Telematics, University of Karlsruhe, Karlsruhe, Germany, 2005, http://doc.tm.uka.de/tr/TM-2005-1.pdf.

[8] S. Cherukuri, K. Venkatasubramanian, and S. Gupta, “Biosec: a biometric based approach for securing communication in wireless networks of biosensors implanted in the human body,” in *Proceedings of the IEEE International Conference on Parallel Processing Workshops*, pp. 432–439, 2003.

[9] C. C. Y. Poon, Y. T. Zhang, and S. D. Bao, “A novel biometrics method to secure wireless body area sensor networks for telemedicine and M-health,” *IEEE Communications Magazine*, vol. 44, no. 4, pp. 73–81, 2006.

[10] K. K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, “EKG-based key agreement in body sensor networks,” in *Proceedings of the IEEE INFOCOM Workshops*, pp. 1–6, IEEE, April 2008.

[11] F. Xu, Z. Qin, C. C. Tan, B. Wang, and Q. Li, “IMDGuard: securing implantable medical devices with the external wearable guardian,” in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM ’11)*, pp. 1862–1870, April 2011.

[12] M. Mana, M. Feham, and B. A. Bensaber, “Trust key management scheme for wireless body area networks,” *International Journal of Network Security*, vol. 12, no. 2, pp. 75–83, 2011.

[13] A. Juels and M. Sudan, “A fuzzy vault scheme,” *Designs, Codes, and Cryptography*, vol. 38, no. 2, pp. 237–257, 2006.

[14] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, “Radio-telepathy: extracting a secret key from an unauthenticated wireless channel,” in *14th Annual International Conference on Mobile Computing and Networking (MobiCom ’08)*, pp. 128–139, ACM, September 2008.

[15] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, “On the effectiveness of secret key extraction from wireless signal strength in real environments,” in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking (MobiCom ’09)*, pp. 321–332, ACM, September 2009.

[16] K. Yazdandoost and K. Sayrafian, “Channel model for body area network (ban), ieee p802. 15-08-0780-09-0006,” Working Group Document IEEE 802.15, 2009.

[17] L. Hanlen, V. Chaganti, B. Gilbert, D. Rodda, T. Lamahewa, and D. Smith, “Open-source testbed for body area networks: 200 Sample/sec, 12 hrs continuous measurement,” in *Proceedings of the IEEE 21st International Symposium on Personal, Indoor and Mobile Radio Communications Workshops (PIMRC ’10)*, pp. 66–71, IEEE, September 2010.

[18] T. S. Rappaport, *Wireless Communications: Principles and Practice*, vol. 2, 1996.

[19] A. Paulraj, R. Nabar, and D. Gore, *Introduction To Space-Time Wireless Communications*, 2003.

[20] D. B. Smith, J. Zhang, L. W. Hanlen, D. Miniutti, D. Rodda, and B. Gilbert, “Temporal correlation of dynamic on-body area radio channel,” *Electronics Letters*, vol. 45, no. 24, pp. 1212–1213, 2009.

[21] S. Ali, V. Sivaraman, and D. Ostry, “Zero reconciliation secret key generation for body-worn health monitoring devices,” in *Proceedings of the 5th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pp. 39–50, ACM, 2012.

[22] T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka, “Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels,” *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 11, pp. 3776–3784, 2005.

[23] M. A. Tope and J. C. McEachen, “Unconditionally secure communications over fading channels,” in *Proceedings of the Communications for Network-Centric Operations: Creating the Information Force (Milcom ’01)*, vol. 1, pp. 54–58, IEEE, October 2001.

[24] L. Shi, M. Li, S. Yu, and J. Yuan, “Bana: body area network authentication exploiting channel characteristics,” in *Proceedings of the 5th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pp. 27–38, ACM, 2012.

[25] S. Mathur, R. Miller, A. Varshavsky, W. Trappe, and N. Mandayam, “ProxiMate: proximity-based secure pairing using ambient wireless signals,” in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, pp. 211–224, ACM, July 2011.

[26] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, “Wireless device identification with radiometric signatures,” in *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking (MobiCom ’08)*, pp. 116–127, ACM, September 2008.

[27] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, “A statistical test suite for random and pseudorandom number generators for cryptographic applications,” DTIC Document, 2001.

[28] H. Liu, J. Yang, Y. Wang, and Y. Chen, “Collaborative secret key extraction leveraging received signal strength in mobile wireless networks,” in *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (IEEE INFOCOM ’12)*, pp. 927–935, IEEE, 2012.

[29] C. E. Shannon and W. Weaver, *A Mathematical Theory of Communication*, 1948.

[30] U. Uludag, S. Pankanti, and A. Jain, “Fuzzy vault for fingerprints,” in *Audio-and Video-Based Biometric Person Authentication*, pp. 55–71, Springer, New York, NY, USA, 2005.

[31] J. E. Hershey, A. A. Hassan, and R. Yarlagadda, “Unconventional cryptographic keying variable management,” *IEEE Transactions on Communications*, vol. 43, no. 1, pp. 3–6, 1995.

[32] U. M. Maurer, “Secret key agreement by public discussion from common information,” *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, 1993.

[33] Q. Wang, H. Su, K. Ren, and K. Kim, “Fast and scalable secret key generation exploiting channel phase randomness in wireless networks,” in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM ’11)*, pp. 1422–1430, IEEE, April 2011.

[34] N. Patwari, J. Croft, S. Jana, and S. K. Kasera, “High-rate uncorrelated bit extraction for shared secret key generation from channel measurements,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 17–30, 2010.

[35] J. Croft, N. Patwari, and S. K. Kasera, “Robust uncorrelated bit extraction methodologies for wireless sensors,” in *Proceedings of the 9th ACM/IEEE International Conference on Information*

Processing in Sensor Networks (IPSN '10), pp. 70–81, ACM, April 2010.

[36] K. Zeng, D. Wu, A. Chan, and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM '10)*, pp. 1–9, March 2010.

[37] M. Wilhelm, I. Martinovic, and J. B. Schmitt, "Secret keys from entangled sensor motes: implementation and analysis," in *Proceedings of the 3rd ACM Conference on Wireless Network Security (WiSec '10)*, pp. 139–144, ACM, March 2010.

[38] L. Yao, S. T. Ali, V. Sivaraman, and D. Ostry, "Decorrelating secret bit extraction via channel hopping in body area networks," in *Proceedings of the IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC '12)*, pp. 1454–1459, IEEE, 2012.

[39] L. Hanlen, D. Smith, J. Zhang, and D. Lewis, "Key-sharing via channel randomness in narrowband body area networks: is everyday movement sufficient?" in *Proceedings of the 4th International Conference on Body Area Networks. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, p. 17, 2009.

[40] L. Yao, S. T. Ali, V. Sivaraman, and D. Ostry, "Improving secret key generation performance for on-body devices," in *Proceedings of the 6th International Conference on Body Area Networks. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, pp. 19–22, 2011.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 216754, 13 pages
http://dx.doi.org/10.1155/2013/216754

Research Article

An Energy-Efficient Key Predistribution Scheme for Secure Wireless Sensor Networks Using Eigenvector

Sung Jin Choi, Kyung Tae Kim, and Hee Yong Youn

College of Information and Communication Engineering, Sungkyunkwan University, Suwon 440-746, Republic of Korea

Correspondence should be addressed to Hee Yong Youn; youn@ece.skku.ac.kr

Received 4 January 2013; Revised 5 May 2013; Accepted 8 May 2013

Academic Editor: Laurence T. Yang

Currently, sensor networks are widely used in various fields. Here secure operations are required for critical applications since the damages are significant if the network is compromised or disrupted. For the security of wireless sensor network, the earlier schemes typically employ asymmetric cryptography. These schemes are, however, often unsuitable for wireless sensor network due to the limited computational power and energy of the sensor nodes. To address this issue, various approaches have been developed, and the random key predistribution approach has been recognized as an effective approach. One shortcoming, however, is that a common key is not guaranteed to be found between any two nodes wanting to communicate. This paper proposes a new robust key predistribution scheme solving this problem, with which the security is not compromised even though the data exchanged between the nodes are tapped by an adversary. This is achieved by using the keys assigned based on the notion of eigenvalue and eigenvector of a square matrix of a pool of keys. Mathematical analysis and computer simulation reveal that the proposed scheme significantly reduces the overhead required for secure connectivity and energy consumption of sensor nodes compared to the existing approaches.

1. Introduction

Wide-spread deployment of sensor networks is quite practical these days. A network of thousands or more sensors allows an efficient solution to various challenging tasks: traffic monitoring, monitoring of building with respect to the structure, fire, and security, military sensing and tracking, distributed measurement of seismic activity, real-time pollution monitoring, wild life monitoring, wild fire tracking, and so forth [1–3]. Energy-aware distributed intelligent data gathering with wireless sensor networks is a hot issue lately due to the emerge of big data paradigm [4].

Wireless sensor networks (WSNs) share several common properties with the traditional wireless networks. Both of them include arrays of nodes that are battery powered, have limited computational capabilities and memory, and rely on intermittent wireless communication via radio frequency and, possibly, optical links. They also include *data-collecting nodes* which cache the sensed data and make them available to the processing components of the network and *control nodes* which monitor the status of the sensor nodes and broadcast simple commands to them. However, WSNs differ from the traditional wireless networks in several aspects; namely, the scale is a few orders of magnitude larger than that of wireless networks; they are dynamic in the sense that addition and removal of sensor nodes are allowed after the deployment to expand the network or replace failed or unreliable nodes without physical contact; and they may be deployed in hostile areas where the communication is monitored and the sensor nodes are subject to capture and manipulation by an adversary. These harsh operational conditions place very critical security constraints on the WSN design [5–9].

Numerous commercial and military applications require secure operation of sensor networks, and seriously detrimental outcomes might be caused if the network is compromised or disrupted. When the sensor networks are deployed in a hostile environment, security is extremely important as they are prone to different types of malicious attacks. For example, an enemy can easily tap the information, imitate one of the sensor nodes, or intentionally provide incorrect information to other nodes. The critical issue here is how to secure the communication between the sensor nodes; that is, how to set up a secret key between the communicating nodes. Most of the earlier schemes use asymmetric cryptography to solve this problem [10]. However, these schemes are often unsuitable to distributed sensor network due to limited computational power and energy of sensor nodes.

To address this issue a scheme has been proposed, which is based on random key pre-distribution. However, it has a shortcoming that a common key is not guaranteed to be found between any two nodes wanting to communicate, and there is also a high possibility of leakage of key information and breakdown of security. This is because the keys are distributed using an identifier, working as a key transport between the sensor nodes. This paper proposes a new key pre-distribution scheme solving this problem by assigning the keys based on the notion of eigenvalues and eigenvectors [11] of a square matrix of a pool of random keys. The main idea here is that there exists infinite combination of eigenvalues and eigenvectors building a matrix. As a result, one cannot ever conjecture the original matrix with only a portion of eigenvalues and eigenvectors of the sensor nodes, which corresponds to the shared key. It thus provides high security to wireless sensor networks of pre-distributed keys by not exposing any data on the key to other nodes. The main advantages and contributions are summarized as follows.

(i) A common key is guaranteed to be found between any two nodes wanting to communicate.

(ii) The key cannot be leaked unless entire sensor nodes are compromised.

(iii) It requires much smaller memory space to hold the pre-distributed keys.

(iv) The energy efficiency is higher.

Analytical modeling and computer simulation reveal that the proposed scheme significantly reduces the overhead required for secure connectivity and energy consumption of the sensor nodes compared to the existing approach employing the random key pre-distribution scheme.

The rest of the paper is organized as follows. Section 2 discusses the existing key distribution approaches for sensor networks, and Section 3 presents the proposed scheme. Section 4 analyzes and compares the performance of the proposed scheme with those of the earlier schemes, and finally concluding remarks are given in Section 5.

2. Related Works

There exist a number of key pre-distribution schemes developed for wireless sensor network. A basic approach is to let all the nodes carry a master secret key, and any pair of nodes use the global master key for key agreement and creation of a new pairwise key. This approach does not exhibit sufficient network resilience such that if one node is compromised, the security of the entire sensor network is compromised. Some existing studies suggest storing the master key in tamper-resistant hardware to reduce the risk [12], but this increases the cost and energy consumption of each sensor node. Furthermore, tamper-resistant hardware might not always be safe. Liu and Ning [13] proposed another key pre-distribution scheme which substantially improves the resilience of the network compared to other schemes. This scheme exhibits a threshold property; when the number of compromised nodes is smaller than the threshold, the probability that other noncompromised nodes are affected is close to zero. This desired property lowers the initial payoff of small-scale network breaches to an adversary, and makes it necessary for the adversary to attack a significant portion of the network.

Blundo et al. [14] proposed several schemes which allow any group of some parties to compute a common key while being secure against collusion between some members of them. These schemes focus on saving communication cost while memory constraints are not placed on the group members. Perrig et al. [10] proposed SPINS, a security architecture specifically designed for sensor networks. In SPINS, each sensor node shares a secret key with the base station, and any pair of sensor nodes cannot directly establish a secret key. They use the base station as a trusted third party to set up a secret key.

Eschenauer and Gligor [15] proposed a random key pre-distribution scheme, which exploits the probabilistic characteristics of random graph. In this scheme the basestation first creates a large number of random keys and saves them in the key pool. Then, a group of keys are randomly selected from it to build a key ring, which is distributed to the sensor nodes. The sensor nodes find the shared keys among the neighboring nodes residing within the wireless communication radius by broadcasting the key ring and key information to the neighbor nodes. Any two nodes apart by two or more links or having no shared key have to create a path key in order to have a shared key. One of the two nodes selects a key from the key ring and transmits it to other nodes through the intermediate nodes in the key path until reaching the target node. The operation of this scheme consists of three phases as follows, which is illustrated in Figure 1.

2.1. Phase I (The Initialization). The Eschenauer and Gligor (E-G) scheme randomly decides a pool of keys, P, out of the key space generated by random graph. In each node k keys are randomly selected from P and stored in the memory. This set of k keys is called the node's key ring. The number of keys in the key pool, $|P|$, is chosen such that two random subsets of size m in P will share at least one key with some probability p.

2.2. Phase II (The Discovery of Shared Key). The nodes perform key discovery to find out shared key from their neighbors. The key discovery is performed by assigning a short identifier to each key prior to deployment and having each node broadcast its set of identifiers. The nodes identifying that they contain a shared key in their key ring can then verify that their neighbor actually holds the key through a challenge response protocol. The shared key then becomes the key for that link.

2.3. Phase III (The Setup of Path Key). The nodes can set up path key with the nodes in their vicinity when they cannot

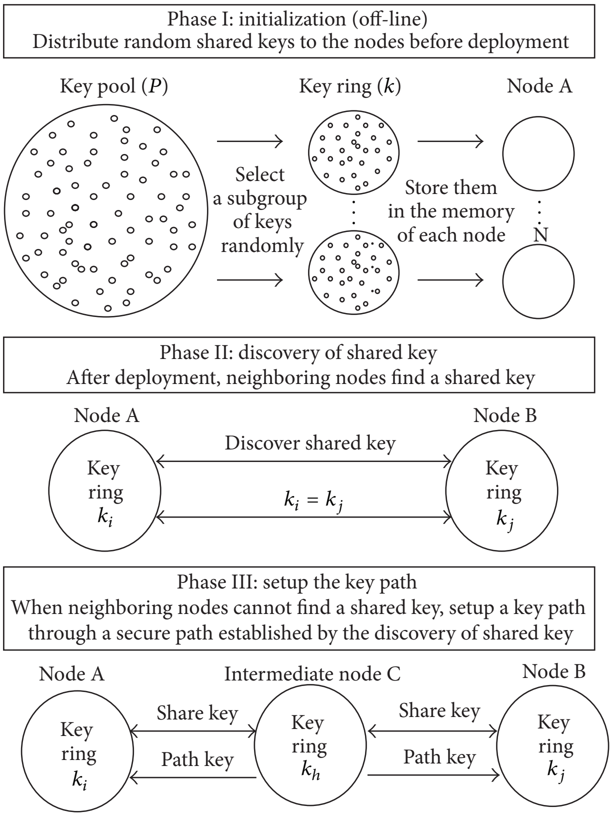

Phase I: initialization (off-line)
Distribute random shared keys to the nodes before deployment
Key pool (P)
Key ring (k)
Node A
Select a subgroup of keys randomly
Store them in the memory of each node
N
Phase II: discovery of shared key
After deployment, neighboring nodes find a shared key
Node A
Node B
Discover shared key
Key ring k_i
$k_i = k_j$
Key ring k_j
Phase III: setup the key path
When neighboring nodes cannot find a shared key, setup a key path through a secure path established by the discovery of shared key
Node A
Intermediate node C
Node B
Share key
Share key
Path key
Path key
Key ring k_i
Key ring k_h
Key ring k_j

Figure 1: The procedure of the Eschenauer and Gligor scheme.

find shared keys from their key rings. If the network is connected, a path can be found from a source node to its neighbor. The source node then generates a path key and sends it securely via the path to the target node.

The key pre-distribution scheme proposed by Chan et al. [16] also employs the random graph approach like the scheme proposed in [15], but it uses q (≥1) shared keys instead of one. This scheme connects two nodes via multiple paths and creates the keys to fortify the security. As a result, even if a sensor node is damaged by the attacker, the security of the rest of the nodes can be preserved using the random/shared keys. This scheme decides a pool of keys of size of P from the key space and composes a key ring of k elements. For the communication of one node with the neighbor nodes, at least more than q keys need to be shared and security is provided by creating a new key ($\text{hash}(k_1|k_2|\cdots k_q)$). This scheme focuses on the security fortification against small-scale attacks. One shortcoming here is that a shared key between any two nodes is not guaranteed to be found. Moreover, it does not support the mechanism for mutual authentication between the nodes.

Camtepe and Yener [17] and Lee and Stinson [18] applied combinatorial approaches to key pre-distribution. They presented two classes of combinatorial designs: symmetric balanced incomplete block designs and generalized quadrangles. The points and blocks in the combinatorial designs are associated with distinct key identifiers and nodes, respectively. Here even though the probability of key establishment has been increased, the network resiliency is still limited and node authentication is not ensured. Sánchez and Baldus [19] made use of combinatorial design theory for the pre-distribution of multiple bivariate polynomial shares based on [14]. Their approach enables direct pairwise key establishment for a large number of nodes, independently of the physical connectivity properties of WSNs. Chakrabarti et al. [20] also used the combinatorial designs for key pre-distribution in WSNs. Their method is to begin with a transversal design and then form the key rings by merging the blocks. Some performance metrics are improved at the cost of larger storage.

Liu et al. [21] proposed an asymmetric key pre-distribution scheme (AKPS). AKPS uses a trusted authority (TA) to distribute secret keys to each user and public keying material to keying material servers (KMSs). With the help of KMSs, two sensor nodes can establish a session key to encrypt messages. AKPS has an advantage over other schemes in that the compromise of KMSs does not disclose any information of the users' secret keys and the session keys. Nguyen et al. proposed a key management scheme considering the signal range in [22]. Each node is assigned with a subset of keys from the key pool by a key setup server. Two nodes residing in each other's communication range is assigned a subset of common keys. This scheme also includes shared key discovery and path key establishment phases. By using the location information of sensor nodes, it improves the connectivity and achieves better resilience than other schemes. However, this scheme depends on the information of sensor deployment.

Szczechowiak and Collier [23] proposed a key agreement scheme based on identity-based cryptograph (IBC) for wireless sensor networks. A trust authority is used to pre-distribute a secret key, a unique identity IDx, a hashing function H, a mapping function, and a key derivation function (KDF) into the memory of each node. This scheme saves much key storage space and allows high resilience against node capture. However, the key agreement protocol employs pairing-based cryptography, which requires large computational and energy resource for each sensor node to compute the shared pairwise keys together with its neighboring nodes.

Some literatures focus on localizing the keys. In [24, 25] the authors presented RPKH and location-dependent key management (LDK) scheme to allow local key management. They utilize different nodes including the normal nodes and anchor nodes to generate the keys of different transmission ranges. The LDK scheme employs heterogeneous sensors to build a clustered sensor network; the higher-ability nodes (anchor nodes) take the management role and regular nodes. The anchor nodes use the location information of other nodes to generate sets of keys. The neighboring nodes establish secure communication link by determining common keys via exchanging the data of their key. LDK takes advantage of relative location of the nodes by utilizing the anchor nodes of different power level. According to the locations, the nodes receive different sets of keys from the anchor nodes, and the neighboring nodes can establish secure communication link through the common keys. LDK can increase the direct connectivity ratio among the nodes. However, the nodes need to transmit a message containing all the data of the key for determining common keys. This operation consumes lots of energy, and thus it is not appropriate for WSNs. Moreover, the adversary can eavesdrop on the exchanged key data, and the anchor nodes are difficult to deploy.

Recently, efficient and secure key management for wireless sensor networks attracted a number of researchers [26–28]. Bechkit et al. [29] and Gu et al. [30] focused on key pre-distribution approach for mainly scalability, and Kim et al. [31] applied the key distribution to the clustered WSN. A new polynomial-based rekeying scheme was also proposed by Guo and Qian [32], and an adaptive dynamic key management approach was proposed by Alcaraz et al. [33]. As a different approach, Yu and Wang [34] and Paterson and Stinson [35] suggested to use combinatorial design. Salam et al. [36] proposed public key cryptography for key pre-distribution. An asymmetric matrix and projective plane was used by Subash and Divya [37] and Mitra et al. [38], respectively. The distinctive features of the proposed scheme compared to these key pre-distribution approaches are that it takes advantages of the notion of eigenvectors of a symmetric matrix, which disallows reverse mapping (and thus compromise of security).

The two main issues in the random key pre-distribution approaches are guaranteeing to find a common key between any pair of nodes and the prevention of information leaks. We next present the proposed scheme effectively handling these issues.

3. The Proposed Scheme

In this section the proposed scheme is presented, deferring the analysis and performance evaluation on the security and energy efficiency to the next section. The proposed key pre-distribution scheme employs the random graph approach like Eschenauer's method [15]. However, it guarantees that any pair of nodes can find the shared key between them while preventing the leakage of key information.

3.1. Preliminaries. The proposed scheme is based on important properties of a matrix in designing the key pre-distribution scheme.

Definition 1 (eigenvalue and eigenvector). Let A be an $n \times n$ matrix. A nonzero vector v is an eigenvector of A if (1) holds for some scalar λ. λ is called an eigenvalue of A corresponding to the eigenvector v. Eigenvalues are also known as characteristic, or proper, values or even as latent roots

$$Av = \lambda v. \tag{1}$$

Let us now discuss how to compute eigenvalues and eigenvectors in general. Because $Av = \lambda v \Rightarrow Av = \lambda I v, \Rightarrow Av - \lambda I v = 0 \Rightarrow (A - \lambda I)v = 0$, we see that v is an eigenvector, if and only if it is a nontrivial solution of the homogeneous system $(A - \lambda I)v = 0$. In this case, v is a nonzero vector of the null space of $A - \lambda I$. The system has a nontrivial solution, if and only if the determinant of the coefficient matrix is zero. Thus, λ is an eigenvalue of A, if and only if $\det(A - \lambda I) = 0$ [39].

Definition 2 (α*-secure*). As long as an adversary compromises less than or equal to α nodes, uncompromised nodes are perfectly secure.

The security of the proposed scheme is stronger than α–secure in the sense that the entire security is unbroken despite $(\alpha + 1)$ nodes being exposed. The entire security could be broken only when the entire keys pre-distributed are leaked, which is virtually impossible.

3.2. The Proposed Key Distribution Scheme. The key pre-distribution scheme proposed in this section randomly selects k keys out of the key pool of p elements and then generates the index of these keys. A random function is used to generate node identifiers, and the keys generated in the key pool are used as session keys.

The session key is an encryption key used for only one communication session. In case a key is used for numerous encryption messages, the key could be extracted from the messages. This is prevented using a temporary session (i.e., one-time) key. The session key approach employed in the earlier schemes may cause key exposure when used repeatedly. This problem is solved by the proposed approach using the pre-distributed key combination (initial vector).

3.2.1. Setup of Initial Vector. The initial vector is set via four off-line steps: (i) generation of a large pool of keys (e.g., $2^{17} \sim 2^{20}$ keys), (ii) formation of a square matrix using the pool of keys, (iii) derivation of eigenvalues and eigenvectors for the square matrix, (iv) and key pre-distribution to each sensor node.

Step 1 (generation of a large pool of keys). The proposed key pre-distribution scheme is based on random keys. Therefore, a large pool of keys (e.g., $2^{17} \sim 2^{20}$ keys) are generated in this step. Each sensor node receives a subset of keys from the pool before deployment. For the communication between two nodes, they need to find one common key to be used as a shared secret key.

Step 2 (forming a square matrix using the pool of keys). Eschenauer's scheme uses just a pool of keys. However, the proposed scheme uses a pool of keys formed in a square matrix. The random keys are first laid out in the square matrix format before applying the proposed key pre-distribution scheme using eigenvalues and eigenvectors.

Step 3 (deriving eigenvalues and eigenvectors for the square matrix). The eigenvalues and eigenvectors derived from the square matrix are stored as keys in each sensor node. It is to allow a common key between any two nodes and increase the security by providing node-to-node mutual authentication.

A general method for finding eigenvalues and eigenvectors shown in Definition 1 needs to be developed. It computes the dominant eigenvalue and eigenvector corresponding to the dominant eigenvalue. Without loss of generality, it is necessary to assume that square matrix, A, has the following two properties.

(i) There is a single eigenvalue of maximum modulus.

(ii) There is a linearly independent set of n eigenvectors.

According to the first assumption, the eigenvalues can be labeled such that

$$\left|\lambda_1\right| > \left|\lambda_2\right| \geq \left|\lambda_3\right| \geq \cdots \geq \left|\lambda_n\right|. \tag{2}$$

According to the second assumption, there is a basis $\{v^{(1)}, v^{(2)}, \ldots, v^{(n)}\}$ for C^n such that

$$A v^{(j)} = \lambda_j v^{(j)} \quad (1 \leq j \leq n). \tag{3}$$

Let $x^{(0)}$ be an element of C^n such that when $x^{(0)}$ is expressed as a linear combination of the basis elements $v^{(1)}, v^{(2)}, \ldots, v^{(n)}$, the coefficient of $v^{(1)}$ is not 0. Thus,

$$x^{(0)} = a_1 v^{(1)} + a_2 v^{(2)} + \cdots + a_n v^{(n)} \quad (a_1 \neq 0). \tag{4}$$

We form then $x^{(1)} = Ax^{(0)}, x^{(2)} = Ax^{(1)}, \ldots, x^{(k)} = Ax^{(k-1)}$ to have

$$x^{(k)} = A^k x^{(0)}. \tag{5}$$

In the following analysis there is no loss of generality in absorbing all the coefficients a_j in the vectors $v^{(j)}$. By (5), we have

$$x^{(k)} = A^k v^{(1)} + A^k v^{(2)} + \cdots + A^k v^{(n)}. \tag{6}$$

Using (3), we arrive at

$$x^{(k)} = \lambda_1^k \left[v^{(1)} + \left(\frac{\lambda_2}{\lambda_1}\right)^k v^{(2)} + \cdots + \left(\frac{\lambda_n}{\lambda_1}\right)^k v^{(n)} \right]. \tag{7}$$

Since $|\lambda_1| > |\lambda_j|$ for $2 \leq j \leq n$, we see that the coefficients $(\lambda_j/\lambda_1)^k$ tend to 0 and the vector within the brackets converges to $v^{(1)}$ as $k \to \infty$.

To simplify the notation, we write $x^{(k)}$ in the form

$$x^{(k)} = \lambda_1^k \left[v^{(1)} + \varepsilon^{(k)} \right] \quad \text{where } \varepsilon^{(k)} \longrightarrow 0 \text{ as } k \longrightarrow \infty. \tag{8}$$

In order to be able to take ratios, let φ be any linear functional on C^n for which $\varphi(v^{(1)}) \neq 0$. Recall that a linear functional φ satisfies $\varphi(\alpha x + \beta y) = \alpha\varphi(x) + \beta\varphi(y)$, for scalars α and β and vectors x and y. (e.g., φ could simply evaluate the jth component of any given vector). Then

$$\varphi\left(x^{(k)}\right) = \lambda_1^k \left[\varphi\left(v^{(1)}\right) + \varphi\left(\varepsilon^{(k)}\right) \right]. \tag{9}$$

Consequently, the following ratios converge to λ_1 as $k \to \infty$:

$$r_k \equiv \frac{\varphi\left(x^{(k+1)}\right)}{\varphi\left(x^{(k)}\right)} = \lambda_1 \left[\frac{\varphi\left(v^{(1)}\right) + \varphi\left(\varepsilon^{(k+1)}\right)}{\varphi\left(v^{(1)}\right) + \varphi\left(\varepsilon^{(k)}\right)} \right] \longrightarrow \lambda_1. \tag{10}$$

Since the direction of the vector $x^{(k)}$ aligns more and more with $v^{(1)}$ as $k \to \infty$, this results in the eigenvector $v^{(1)}$. If the eigenvectors found are

$$v^{(1)} = \begin{bmatrix} s_1 \\ \vdots \\ \vdots \\ s_n \end{bmatrix} s_n \in R, \qquad v^{(2)} = \begin{bmatrix} t_1 \\ \vdots \\ \vdots \\ t_n \end{bmatrix} t_n \in R, \ldots,$$
$$v^{(n)} = \begin{bmatrix} z_1 \\ \vdots \\ \vdots \\ z_n \end{bmatrix} z_n \in R, \tag{11}$$

P ($P = [v^{(1)} v^{(2)} \cdots v^{(n)}]$) is actually stored key and D matrix consisting of eigenvalues becomes the decryption key. An important property of the proposed scheme is that it allows both key pre-distribution and encryption of data at the same time. That is, one cannot extract the original data even though some elements of the P matrix stored in each sensor node are leaked. This is because deciding the original matrix using a small portion of P matrix is impossible. As a result, the proposed key pre-distribution scheme allows high security.

Step 4 (key pre-distribution). In this step every node is assigned P matrix consisting of eigenvectors and D matrix consisting of eigenvalues. Note here that there exist infinite ways for forming P matrix by arbitrarily deciding the s_n, t_n, and u_n values. In addition, only the diagonal elements (eigenvalues) from the D matrix are stored to minimize the required memory space.

3.2.2. Distribution of Session Key.

The previous subsection explained how to generate the initial vector using pre-distributed key combinations. This subsection describes how to distribute the session keys from a source node to the destination node using the initial vector. It consists of four steps: (i) set the initial vector between two nodes, (ii) exchange messages for setting a session, (iii) set the session key, (iv) and update the initial vector.

Step 1 (set the initial vector between two nodes). An initial vector needs to be set between two nodes for which a security session is to be set. This step needs Definition 3.

Definition 3 (eigenvalue and eigenvector). Assume that an $n \times n$ matrix A can be converted to a diagonal matrix D, which is called diagonalizable. Then there exists an invertible $n \times n$ matrix P such that

$$P^{-1} A P = D. \tag{12}$$

The process of finding matrices P and D is called diagonalization. First, it is worth noticing that if D is a diagonal matrix with diagonal entries $\lambda_1, \ldots, \lambda_n$, then for $i = 1, \ldots, n$

$$D e_i = \lambda e_i. \tag{13}$$

Hence, the standard basis vectors $e_1, \ldots, e_n$ are eigenvectors of D. In particular, the eigenvectors of D are linearly independent. To find a common key A using Definition 3, the following theorem needs to be proved.

Theorem 4. *One has the following.*

(i) *A is diagonalizable if and only if it has n linearly independent eigenvectors.*

(ii) *If A is diagonalizable with $P^{-1}AP = D$, then the columns of P are eigenvectors of A and the diagonal entries of D are the corresponding eigenvalues.*

(iii) *If $\{v_1, \ldots, v_n\}$ are linearly independent eigenvectors of A with corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$, then A can be diagonalized by*

$$P = \left[v_1 v_2 \cdots v_n\right], \qquad D = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix}. \tag{14}$$

Proof. Let P be a matrix with columns of n-vectors $v_1, \ldots, v_n$ and let D be a diagonal matrix with diagonal elements, $\lambda_1, \ldots, \lambda_n$, respectively. Then

$$\begin{aligned} AP = A\left[v_1 v_2 \cdots v_n\right] &= \left[v_1 v_2 \cdots v_n\right] \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \\ &= PD. \end{aligned} \tag{15}$$

If A is diagonalizable, with $P^{-1}AP = D$, then $AP = PD$. Hence, $Av_i = \lambda_i v_i$, $i = 1, \ldots, n$. So, the λ_is are eigenvalues and the v_is are corresponding eigenvectors. □

Suppose that A has n linearly independent eigenvectors, say, $v_1 v_2 \cdots v_n$ (the columns of P). If $\lambda_1, \ldots, \lambda_n$ are the corresponding eigenvalues, then $Av_i = \lambda_i v_i, i = 1, \ldots, n$. If D is a diagonal matrix with diagonal entries $\lambda_1, \ldots, \lambda_n$, then $AP = PD$ by (15). Because P is a square matrix with linearly independent columns, it is invertible. Hence, $P^{-1}AP = D$, and A is diagonalizable [40].

If we multiply P and P^{-1} matrix to both sides of (12), it becomes $PP^{-1}APP^{-1} = PDP^{-1}$. Since $PP^{-1} = I$, $P^{-1}P = I$ (I: identity matrix),

$$\begin{aligned} A &= PDP^{-1} \\ &= \left[v^{(1)} \cdots v^{(n)}\right] \begin{bmatrix} \lambda_1 & \cdots & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & \lambda_n \end{bmatrix} \left[v^{(1)} \cdots v^{(n)}\right]^{-1}. \end{aligned} \tag{16}$$

Therefore, the common key can be found from P and D matrix that was stored in the sensor nodes by $A = PDP^{-1}$.

Assume that node$_x$ and node$_y$ contain $(v_1^1, v_1^2 \ldots, v_1^n)$ and $(v_2^1, v_2^2, \ldots, v_2^n)$, respectively. When node$_x$ and node$_y$ need to find the initial vector between them, they first exchange randomly selected vectors and then compute a vector product as follows;

$$\begin{aligned} &\text{node}_x : [v_1^1 \cdots v_2^k \cdots v_1^n][\lambda_1 \cdots \lambda_n][v_1^1 \cdots v_2^k \cdots v_1^n]^{-1}, \\ &\qquad \text{node}_y\text{:} [v_2^1 \cdots v_1^k \cdots v_2^n][\lambda_1 \cdots \lambda_n][v_2^1 \cdots v_1^k \cdots v_2^n]^{-1}. \end{aligned}$$

Recall that A is diagonalizable, and thus A is used as an initial vector between node$_x$ and node$_y$.

Step 2 (exchange messages for setting a session). Using the pre-distributed key combinations as the initial vector, m keys from the k keys are selected at each node. Also, $T_{ck(i)}$ in the source node and $D_{ck(i)}$ in the destination node are generated by applying Exclusive-OR operation to the key selected. Using $T_{ck(i)}$, $D_{ck(i)}$, and the initial vector $SD_{\mathrm{IV}(k)}$, an encrypted message required for setting a session is exchanged between the nodes. They are MS $(= SD_{\mathrm{IV}(k)} \oplus T_{ck(i)})$ and MD $(= SD_{\mathrm{IV}(k)} \oplus D_{ck(i)})$.

Step 3 (set the session key). In this step, with the messages exchanged in Step 2 used to set a session at each node, the session key is generated. The message exchanged to set the session extracts $T_{ck(i)}$ and $D_{ck(i)}$ with the Exclusive-OR operation, and each node generates $SD_{sk(i)} = MS \oplus MD = T_{ck(i)} \oplus D_{ck(i)}$ as the session key.

Step 4 (update initial vector). A secure communication is available at each node using the session key decided in Step 3. In this step the initial vector is updated to improve the security level at each node. The updated initial vector $SD_{\mathrm{IV}(i+1)}$ is extracted by the messages setting the session, MS and MD, and the initial vector, $SD_{\mathrm{IV}(i)}$, as in the following expression:

$$SD_{\mathrm{IV}(k+1)} = MS \oplus MD \oplus SD_{\mathrm{IV}(k)}. \tag{17}$$

The two nodes now share $SD_{sk(i)}$ as the session key. Figure 2 depicts the flow of message exchange between the two communicating nodes.

3.3. Example

3.3.1. Setup of Initial Vector

Step 1 (generation of a large pool of keys).

Step 2 (forming a square matrix using the pool of keys). We have

$$\begin{bmatrix} 3 & 0 & 0 \\ -4 & 6 & 2 \\ 16 & -15 & -5 \end{bmatrix}. \tag{18}$$

Step 3 (deriving eigenvalues and eigenvectors for the square matrix).

The eigenvalues obtained are $\lambda_1 = 0, \lambda_2 = 1$, and $\lambda_3 = 3$. Eigenvectors corresponding to each λ become

$$\begin{gathered} v^1 = \begin{bmatrix} 0 \\ s \\ -3s \end{bmatrix}, \qquad v^2 = \begin{bmatrix} 0 \\ 2t \\ -5t \end{bmatrix}, \qquad v^3 = \begin{bmatrix} u \\ 0 \\ 2u \end{bmatrix} \\ s, t, u \in R. \end{gathered} \tag{19}$$

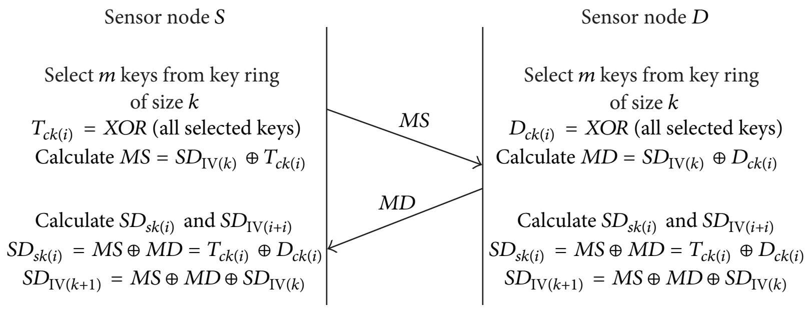

Sensor node S
Sensor node D
Select m keys from key ring of size k
$T_{ck(i)} = XOR$ (all selected keys)
Calculate $MS = SD_{\mathrm{IV}(k)} \oplus T_{ck(i)}$
MS
Select m keys from key ring of size k
$D_{ck(i)} = XOR$ (all selected keys)
Calculate $MD = SD_{\mathrm{IV}(k)} \oplus D_{ck(i)}$
MD
Calculate $SD_{sk(i)}$ and $SD_{\mathrm{IV}(i+i)}$
$SD_{sk(i)} = MS \oplus MD = T_{ck(i)} \oplus D_{ck(i)}$
$SD_{\mathrm{IV}(k+1)} = MS \oplus MD \oplus SD_{\mathrm{IV}(k)}$
Calculate $SD_{sk(i)}$ and $SD_{\mathrm{IV}(i+i)}$
$SD_{sk(i)} = MS \oplus MD = T_{ck(i)} \oplus D_{ck(i)}$
$SD_{\mathrm{IV}(k+1)} = MS \oplus MD \oplus SD_{\mathrm{IV}(k)}$

FIGURE 2: The flow of message exchange between the nodes.

Therefore, the actually stored key P and decryption key D are as follows:

$$P = \left[v^1, v^2, v^3\right], \qquad D = \left[0\ \ 1\ \ 3\right]. \tag{20}$$

Step 4 (key pre-distribution). We have

$$\begin{gathered}
\mathrm{node}_x : P = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 2 & 0 \\ -3 & -5 & 2 \end{bmatrix}, \qquad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \\
s = t = u = 1, \\
\mathrm{node}_y : P = \begin{bmatrix} 0 & 0 & 2 \\ 1 & -2 & 0 \\ -3 & 5 & 4 \end{bmatrix}, \qquad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \\
s = 1, t = -1, u = 2.
\end{gathered} \tag{21}$$

3.3.2. Distribution of Session Key

Step 1 (set the initial vector between two nodes). When node_x and node_y need to find the initial vector between them, they first exchange randomly selected vectors v^3 (v^3 of the node_x is [1 0 2], v^3 of the node_y is [2 0 4]) and then compute a vector product as follows:

$$\begin{gathered}
\mathrm{node}_x : \begin{bmatrix} 0 & 0 & 2 \\ 1 & 2 & 0 \\ -3 & -5 & 4 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 0 & 0 & 2 \\ 1 & 2 & 0 \\ -3 & -5 & 4 \end{bmatrix}^{-1} \\
= \begin{bmatrix} 3 & 0 & 0 \\ -4 & 6 & 2 \\ 16 & -15 & -5 \end{bmatrix} = SD_{\mathrm{IV}(1)}, \\
\mathrm{node}_y : \begin{bmatrix} 0 & 0 & 1 \\ 1 & -2 & 0 \\ -3 & 5 & 2 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 1 & -2 & 0 \\ -3 & 5 & 2 \end{bmatrix}^{-1} \\
= \begin{bmatrix} 3 & 0 & 0 \\ -4 & 6 & 2 \\ 16 & -15 & -5 \end{bmatrix} = SD_{\mathrm{IV}(1)}.
\end{gathered} \tag{22}$$

Step 2 (exchange message for setting a session). We have

$T_{ck(3)} = [1\ 0\ 2]$ (3 rd column vector of the node_x), $SD_{\mathrm{IV}(1)} = [-15\ \ -5\ \ -4\ \ 0\ \ 2\ \ 3\ \ 6\ \ 16]$

$MS = SD_{\mathrm{IV}(1)} \oplus T_{ck(3)} = [-15\ \ -5\ \ -4\ \ 0\ \ 1\ \ 2\ \ 3\ \ 6\ \ 16]$

$D_{ck(3)} = [2\ 0\ 4]$ (3 rd cloumn vector of the node_y), $SD_{\mathrm{IV}(1)} = [-15\ \ -5\ \ -4\ \ 0\ \ 2\ \ 3\ \ 6\ \ 16]$

$MD = SD_{\mathrm{IV}(1)} \oplus D_{ck(3)} = [-15\ \ -5\ \ -4\ \ 0\ \ 2\ \ 3\ \ 4\ \ 6\ \ 16]$.

Step 3 (set the session key). $SD_{\mathrm{IV}(1)} = MS \oplus MD = [-15\ \ -5\ \ -4\ \ 0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 6\ \ 16]$.

Step 4 (update initial vector). $SD_{\mathrm{IV}(2)} = MS \oplus MD \oplus SD_{\mathrm{IV}(1)} = [-5\ \ -5\ \ -4\ \ 0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 6\ \ 16]$.

4. Performance Evaluation

4.1. Analysis of Connectivity. A random graph $G(n, p)$ is a graph of n nodes for which the probability that a link exists between two nodes is p. When p is zero, the graph does not have any edge, whereas when p is one, the graph is fully connected. Spencer [41] and Erdős and Rényi [42] showed that, for monotone properties, there exists a value of p such that the property moves from "nonexistent" to "certainly true" in a very large random graph. The function defining p is called the threshold function of the property. Given a desired probability P_c for graph connectivity, the threshold function p is defined by

$$\begin{gathered}
P_c = \lim_{n \to \infty} P_r \left[G(n, p) \text{ is connected}\right] = e^{e^{-c}}, \\
\text{where } p = \frac{\ln(n) - \ln(-\ln(P_c))}{n}.
\end{gathered} \tag{23}$$

Let p be the probability that a shared key exists between two sensor nodes, and let n be the number of nodes, and d be the expected degree as

$$d = p \times (n - 1) = \frac{(n - 1)(\ln(n) - \ln(-\ln(P_c)))}{n}. \tag{24}$$

For the deployment of a sensor network, let N be the expected number of neighbors within the communication range of a node. Using the expected node degree discussed above, the required local connectivity, P_{required}, can be estimated as follows:

$$P_{\text{required}} = \frac{d}{N} = \frac{(n - 1)(\ln(n) - \ln(-\ln(P_c)))}{nN}. \tag{25}$$

After the required local connectivity is obtained, the value S (the size of the key pool) and k (the number of keys in each node) are decided. Note that S is not directly related to the sensor network, while k is related to the memory size of a sensor node. Therefore, k needs to be as small as possible. Denote P_{actual}, the actual local connectivity, which is the probability of any two neighboring nodes to find a common key between themselves. The link availability of any two nodes of the existing schemes [15, 16] is obtained by $1-P_{\text{ns}}$, where P_{ns} is the probability that a pair of nodes do not share a common key. It can be found using P_{actual},

$$P_{\text{actual}} = 1 - \frac{{}_{S}C_{k} \times {}_{S-k}C_{k}}{\left({}_{S}C_{k}\right)^{2}} = 1 - \frac{((S-k)!)^{2}}{S!\,(S-2k)!}. \tag{26}$$

P_{actual} is then approximated as follows:

$$P_{\text{actual}} = 1 - \frac{(S-k)^{2S-2k+1}}{(S-2k)^{S-2k+(1/2)}}. \tag{27}$$

Recall that the Eschenauer and Gligor (E-G) [15] and Chan et al. (C-P) [16] schemes are the two representative key predistributions employing the random graph approach like the proposed scheme. Therefore, the efficiency of the proposed scheme is compared with these two schemes. Figure 3 compares the actual local connectivity of the proposed scheme with them when the size of the key varies from 2 to 200 for the S value of 5000 and 10000. Observe from the figure that the local connectivity increases as the number of keys in a node increases for the existing scheme when the size of the pool of keys is fixed. Note that the proposed scheme always allows the connectivity regardless of the number of keys per node. Also, the superiority of the proposed scheme becomes more substantial when the memory size of a sensor node is small.

4.2. Security of Connection. When the sizes of the key pool and key ring are S and K, respectively, the probability of two key rings sharing at least one key is calculated by (27). The number of cases each of two nodes chooses k keys out of the entire key pool is estimated in (28)

$$(C(S,k))^{2} = \left(\frac{S!}{k!\,(S-k)!}\right)^{2}. \tag{28}$$

Equation (29) estimates the number of cases where i shared keys are chosen from the entire key pool

$$C(S,i) = \frac{S!}{(S-i)!i!}. \tag{29}$$

The probability of two nodes sharing exactly i keys is calculated in (30)

$$P(i) = \frac{C(S,i)\,C(S-i,2(k-i))\,C(2(k-i),k-i)}{(C(S,k))^{2}}. \tag{30}$$

If a node of a wireless sensor network is captured by an adversary, the key information kept in the node might be exposed. The degree that the sensor nodes can operate while maintaining a desired security level is defined as the resilience against node capture. The higher the resilience against node capture, the longer the security level can be maintained during the operation of wireless sensor network.

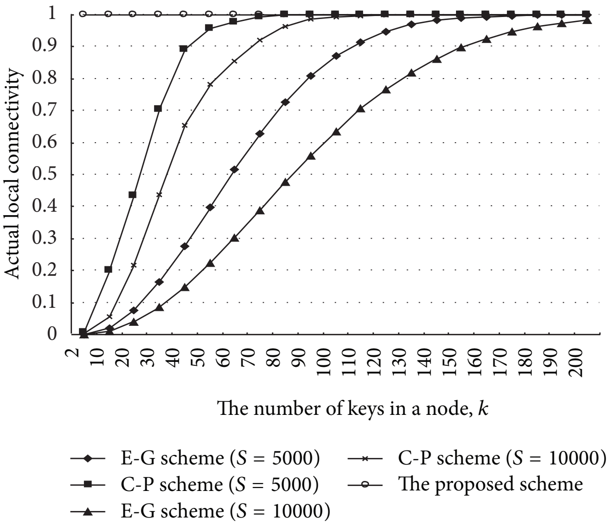


Figure 3: Comparison of connectivity of different schemes.

In the proposed scheme, the number of keys used for session key distribution is α, which is much larger than i, leading to a lower probability of a communication link being exposed, compared to the existing schemes. Equation (31) shows that the probability of the session key to be exposed leads to a large value when x number of nodes are captured with the proposed scheme

$$\left(1-\left(1-\frac{k}{S}\right)^{x}\right)^{\alpha}, \quad i < \alpha \leq k. \tag{31}$$

The number of neighbor nodes sharing a key with the existing random graph based schemes is obtained by simulation to evaluate the security of the connection depending on the sizes of key rings. In the simulation the size of key rings is reduced in units of 100 each time, starting from 1,000 to 2,000, where the size of the key pool is 100,000. Then, the probability of establishing a secure connection between any source and destination node randomly selected is found by simulation. In addition, the hop count of the path required for secure connection is found to evaluate the efficiency of the scheme.

Figure 4 shows the average number of neighbor nodes as the size of key ring a node has changed, when the size of the key pool is 100,000 and 1,000 nodes deployed with a 50 m transmission zone in a $1,000 \times 1,000$ region. The number of neighbor nodes with the existing schemes represents the number of nodes sharing a key, whereas it does the ones which can communicate with each other with the proposed approach. Note that the proposed approach is not affected by the size of the key ring. Observe from the figure that a key is shared when the size of key ring exceeds 600 (E-G scheme) and 500 (C-P scheme), respectively. It is predicted that the security of the connection degrades as the size of the key ring

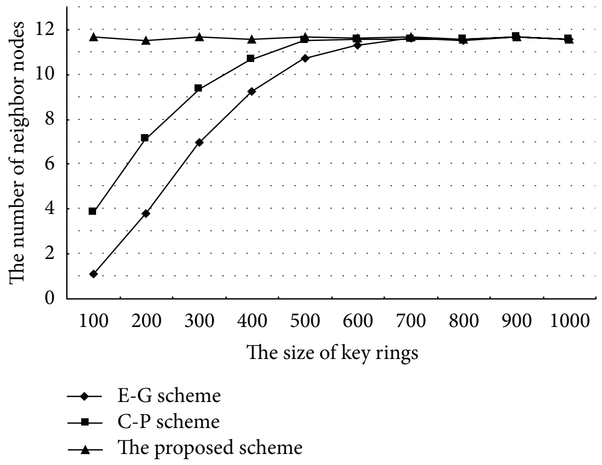


Figure 4: Comparison of the number of neighbor nodes sharing a key.

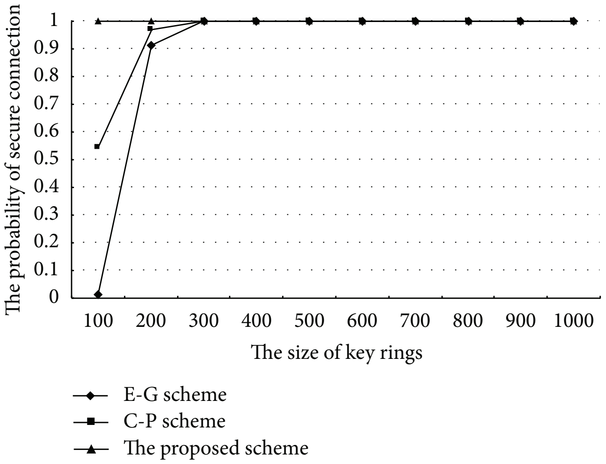


Figure 5: Comparison of the probability of secure connectivity depending on size of key ring.

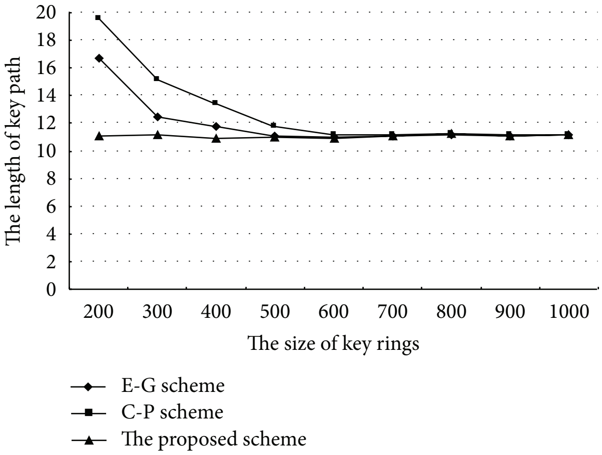


Figure 6: Comparison of key path length depending on the size of key ring.

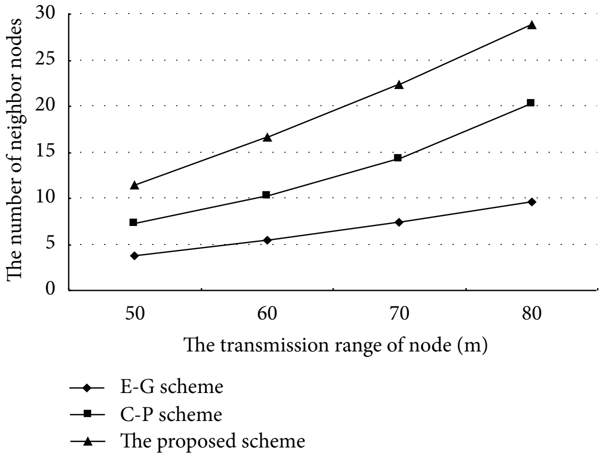


Figure 7: The number of neighbor nodes versus transmission range.

becomes smaller, causing the number of neighbor nodes to decrease dramatically.

Figure 5 shows the result of the evaluation of secure connectivity as the size of the key ring varies. It tests if secure connection is allowed between two randomly selected nodes. The proposed scheme always guarantees secure connectivity regardless of the size of the key ring, while the previous schemes do it only when the size exceeds around 300. The probability of secure connectivity drops sharply when the size falls below 300, for which the number of neighbor nodes is small.

Figure 6 shows the average length of the key path as the size of key ring varies. The length of the key path measures the distance from a node to the destination node where secure connection is allowed. When the size of a key ring is 100, the probability of secure connectivity is close to 0, and thus it is excluded from the test. The proposed scheme demonstrates consistent size of key path irrespective of the size of the key ring, whereas the existing schemes show similar performance to that of the proposed scheme when the size of the key ring exceeds 500. However, with the existing scheme, the lengths increase if the size of the key ring drops below 400.

Figure 7 shows the number of neighbor nodes as the transmission range of a node changes. As the transmission range increases, the number of neighbor nodes important for secure connectivity rises. This indicates that probability of secure connectivity grows in proportion to the transmission range. Figure 8 shows the length of the key path as the transmission range of a node varies. Observe from the figure that the difference between the proposed scheme and the previous ones decreases as the transmission range increases. That is, the increased transmission range reduces the overhead of providing secure connectivity and the required key. However, increasing the transmission range significantly increases energy consumption.

The neighbor nodes of a node can include the ones of multiple hop away to improve the probability of secure connectivity. However, it may increase the energy overhead in

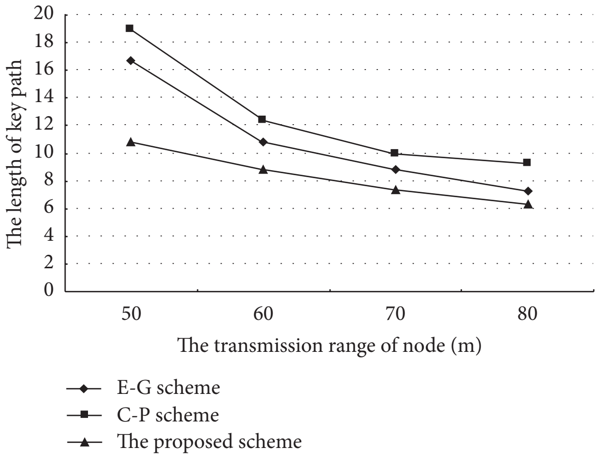


Figure 8: Comparison of key path length versus transmission range.

setting the keys and the number of nodes involved in setting the path key, resulting in lower security.

4.3. Energy Efficiency. We evaluate the energy efficiency of the proposed scheme through computer simulation by applying it to the LEACH protocol [43], which is one of the representative routing protocols proposed for WSNs. The number of cluster heads in the simulation is about 5% of the total number of nodes. We consider a sensor network of 100 sensor nodes randomly arranged in a 100 × 100 region. A base station is located at (50, 50).

In the simulation E_{elec} of the transmitter or receiver circuitry and ε_{amp} of the transmitter amplifier are set to 50 nJ/bit and 100 pJ/bit/m^2 [44–46], respectively, and initial residual energy of a sensor node is set to 25 J. The size of data packet is 3000 bits, the number of key rings of [15, 16] is 100, and key length is 32 bits. The energy consumption model [44, 46] is described as follows. For transmission, when a node transmits k-bit data to another node with distance d between them, the energy it consumes is

$$E_{\text{Tx}}(k, d) = E_{\text{elec}} \times k + \varepsilon_{\text{amp}} \times k \times d^2. \tag{32}$$

For receiving, when a node receives k-bit data, the energy it consumes is

$$E_{\text{Rx}}(k) = E_{\text{elec}} \times k. \tag{33}$$

The parameters used in the simulation are summarized in Table 1.

In the existing schemes each sensor node finds the neighboring nodes having the shared key for which energy consumption is not large. However, the energy consumption increases with the broadcasting of the identifier of the destination node via neighbor nodes holding a shared key for searching the pass key. If a node creates a pass key once and the session key is used more than once, the energy consumption may not greatly increase.

Table 1: The parameters used in the simulation.

Parameters	Value
Network size	100 × 100
Location of base station	(50, 50)
Number of nodes	100
Initial energy of the node	25 J
E_{elec}	50 nJ/bit
ε_{amp}	100 pJ/bit/m^2
Data size	3000 bits
Number of key rings	100
Key length	32 bits

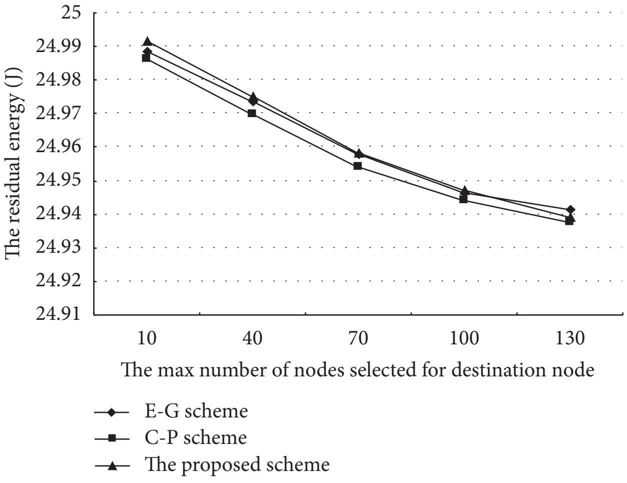


Figure 9: Comparison of the residual energy of the schemes.

Figure 9 compares the energy consumption when a node is randomly selected as the destination node out of some number of nodes (called destination group). Here only the energy used by the transceiver of a node taken for secure connection is considered. The energy consumed by the proposed scheme and previous schemes is low if the destination group is small. It gradually increases as the destination group increases. In the proposed scheme this is due to the increment of energy consumption taken to exchange the initial vector, while it owes to the energy consumed to set up the pass key in the previous scheme. The proposed scheme is affected by the size of destination group more significantly than the previous scheme, but it consumes less energy than the previous scheme in a typical environment where the size of destination group is usually small.

Figure 10 evaluates the energy consumption as time varies when the size of destination group is set to 100. The entire energy consumption is affected by the energy used to distribute the session key and the efficiency of the distribution of the session key. The increment of the energy consumption owes to session key distribution via the pass key in case of the previous schemes. The session key distribution of the previous scheme consumes more energy than the proposed scheme because the path for the pass key is longer.

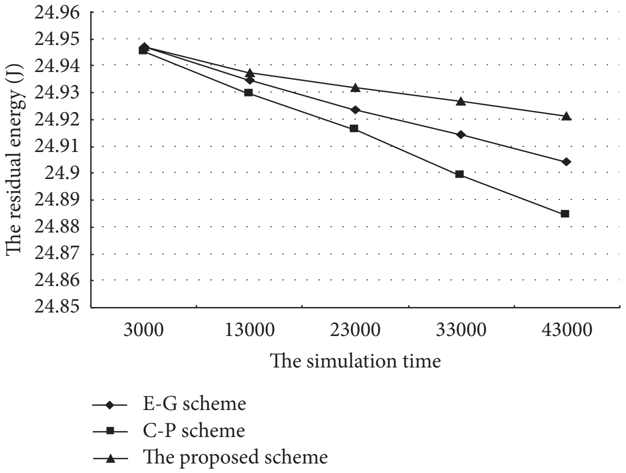


Figure 10: Comparison of the residual energy of the schemes as the key length varies.

5. Conclusion and Future Work

Most earlier schemes proposed for the security of wireless network used asymmetric cryptography such as the Diffie-Hellman key agreement or RSA. However, these schemes are inappropriate for wireless sensor networks due to the limited computation and energy resource of sensor nodes. In order to solve the problem the key distribution scheme using the trusted server was proposed based on asymmetric public key certification approach. Also, the key pre-distribution scheme that saves the key information before installing the sensor node was proposed, which is known to be very effective. The key pre-distribution scheme proposed by Eschenauer and Gligor creates various random keys in the base station, and the randomly selected keys are distributed to each sensor node. The sensor nodes having a common key between them use it as a mutual secret key. When they cannot create a path key, in other words when they cannot find a secret key, they cannot communicate with each other.

This paper has proposed a new key pre-distribution scheme guaranteeing that any pair of nodes can find a common secret key between themselves by using the keys assigned by eigenvalue and eigenvector of a square matrix of a pool of keys. Mathematical analysis and computer simulation revealed that the proposed scheme significantly reduces the overhead required for secure connectivity and energy consumption compared to the existing schemes. Analysis shows that the existing scheme requires a large number of keys in each sensor node to display a comparable connectivity as the proposed scheme. The probability of secure connectivity was evaluated by computer simulation for a randomly designated destination node, which reveals that the size of key ring of the existing schemes needs to be over 300 to be comparable with the proposed scheme. Also, the number of neighbor nodes having a shared key with the existing schemes needs to be over 600. The superiority of the proposed scheme is more substantial when the memory size of the sensor node is small.

When composing a network, the keys need to be pre-distributed as the nodes are deployed in the field. The effectiveness of the key pre-distribution scheme needs to be analyzed as new nodes are added. This is because the probability of secure connectivity may change as the network topology is varied. There is a need of application and performance analysis for the distributed sensor network model covering various conditions including the size of network. In order to raise the probability of secure connectivity, the transmission range of a node can be increased. However, the energy cost increases in proportion to the distance. A new approach considering the energy efficiency along with secure connectivity will also be investigated in the future. The proposed scheme capitalizes some important properties of matrix including eigenvalues and eigenvectors in generating a pool of random keys. It will be expanded to exploit some other properties which allow higher security at lower implementation and operation cost.

Acknowledgments

This research was supported in part by Korea Association of Industry, Academy and Research Institute (C0017380), DAPA and ADD (UD10070MD), Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1 A2040257), and the MSIP (Ministry of Science, ICT and Future Planning), Korea, in the ICT R&D Program 2013.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] D. Du, H. Xiong, and H. Wang, "An efficient key management scheme for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 406254, 14 pages, 2012.

[3] S. Ruj and B. Roy, "Revisiting key predistribution using transversal designs for a grid-based deployment scheme," *International Journal of Distributed Sensor Networks*, vol. 5, no. 6, pp. 660–674, 2009.

[4] R. Zhu, Y. Qin, and J. Wang, "Energy-aware distributed intelligent data gathering algorithm in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 235724, 13 pages, 2011.

[5] R. kishore, S. Radha, and S. G. Hymlin Rose, "Wireless sensor network survey," *International Journal of Distributed Sensor Networks*, vol. 5, no. 6, pp. 850–866, 2009.

[6] A. Goyal, N. Kaur, Padmavati, Kuldeep, and R. Garimella, "Distributed energy efficient key distribution for dense wireless sensor networks," in *Proceedings of the 1st International Conference on Computational Intelligence, Communication Systems and Networks (CICSYN '09)*, pp. 143–148, July 2009.

[7] D. Liu, P. Ning, A. Liu, C. Wang, and W. K. Du, "Attack-resistant location estimation in wireless sensor networks," *ACM Transactions on Information and System Security*, vol. 11, no. 4, pp. 1–22, 2008.

[8] A. Boukercha, L. Xua, and K. EL-Khatibb, "Trust-based security for wireless ad hoc and sensor networks," *Computer Communications*, vol. 30, no. 11-12, pp. 2413–2427, 2007.

[9] B. C. Neuman and T. Ts'o, "Kerberos. An authentication service for computer networks," *IEEE Communications Magazine*, vol. 32, no. 9, pp. 33–38, 1994.

[10] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: security protocols for sensor networks," in *Proceedings of the 7th ACM/IEEE Annual International Conference on Mobile Computing and Networking*, pp. 189–199, July 2001.

[11] S. J. Choi, H. Y. Youn, and B. K. Lee, "An efficient dispersal and encryption scheme for secure distributed information storage," in *Proceedings of the International Conference on Computational Science*, pp. 958–967, 2003.

[12] R. Anderson and M. Kuhn, "Tamper resistance: a cautionary note," in *Proceedings of the 2nd Usenix Workshop on Electronic Commerce*, vol. 2, pp. 1–11, 2008.

[13] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 52–61, 2003.

[14] C. Blundo, A. D. Santix, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology*, pp. 471–486, 1993.

[15] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November 2002.

[16] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 197–213, May 2003.

[17] S. A. Camtepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor network," in *Proceedings of the 9th European Symposium on Research in Computer Security (ESORICS '04)*, pp. 293–308, 2004.

[18] J. Lee and D. R. Stinson, "On the construction of practical key predistribution schemes for distributed sensor networks using combinatorial designs," *ACM Transactions on Information and System Security*, vol. 11, no. 2, pp. 1–35, 2008.

[19] D. Sánchez and H. Baldus, "A deterministic pairwise key predistribution scheme for mobile sensor networks," in *Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks (SecureComm '05)*, pp. 277–288, September 2005.

[20] D. Chakrabarti, S. Maitra, and B. Roy, "A key pre-distribution scheme for wireless sensor networks: merging blocks in combinatorial design," *International Journal of Information Security*, vol. 5, no. 2, pp. 105–114, 2006.

[21] Z. Liu, J. Ma, Q. Huang, and S. Moon, "Asymmetric key predistribution scheme for sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1366–1372, 2009.

[22] H. T. T. Nguyen, M. Guizani, M. Jo, and E. N. Huh, "An efficient signal-range-based probabilistic key predistribution scheme in a wireless sensor network," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 5, pp. 2482–2497, 2009.

[23] P. Szczechowiak and M. Collier, "Practical identity-based key agreement for secure communication in sensor networks," in *Proceedings of the 18th International Conference on Computer Communications and Networks (ICCCN '09)*, pp. 1–6, August 2009.

[24] S. Banihashemian and A. G. Bafghi, "A new key management scheme in heterogeneous wireless sensor networks," in *Proceedings of the 12th International Conference on Advanced Communication Technology (ICACT '10)*, pp. 141–146, February 2010.

[25] F. Anjum, "Location dependent key management in sensor networks without using deployment knowledge," *Wireless Networks*, vol. 16, no. 6, pp. 1587–1600, 2010.

[26] A. Diop, Y. Qi, Q. Wang, and S. Hussain, "An efficient and secure key management scheme for hierarchical wireless sensor networks," *International Journal of Computer and Communication Engineering*, vol. 1, no. 4, pp. 365–370, 2012.

[27] X. He, M. Niedermeier, and H. Meer, "Dynamic key management in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 611–622, 2013.

[28] M. A. Simplicio Jr., P. S. L. M. Barreto, C. B. Margi, and T. C. M. B. Carvalho, "A survey on key management mechanisms for distributed wireless sensor networks," *Computer Networks*, vol. 54, no. 15, pp. 2591–2612, 2010.

[29] W. Bechkit, Y. Challal, A. Bouabdallah, and V. Tarokh, "A highly scalable key pre-distribution scheme for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 948–959, 2013.

[30] W. Gu, X. Bai, and S. Chellappan, "Scaling laws of key predistribution protocols in wireless sensor networks," Tech. Rep., The Department of Computer Science, Missouri University of Science and Technology, 2010.

[31] J. Kim, J. Lee, and K. Rim, "Energy efficient key management protocol in wireless sensor networks," *International Journal of Security and Its Applications*, vol. 4, no. 2, pp. 1–12, 2010.

[32] S. Guo and Z. Qian, "A compromise resilient pair wise rekeying protocol in large scale wireless sensor networks," in *Smart Wireless Sensor Networks*, vol. 18, pp. 316–326, InTechOpen, 2010.

[33] C. Alcaraz, J. Lopez, R. Roman, and H. Chen, "Selecting key management schemes for WSN applications," *Computers & Security*, vol. 31, no. 8, pp. 956–966, 2012.

[34] W. Yu and S. Wang, "Key pre-distribution using combinatorial designs for wireless sensor networks," *WSEAS Transactions on Mathematics*, vol. 12, no. 1, pp. 32–41, 2013.

[35] M. B. Paterson and D. R. Stinson, "A unified approach to combinatorial key predistribution schemes for sensor networks," *Designs Codes and Cryptography*, pp. 1–27, 2012.

[36] M. I. Salam, P. Kumar, and H. Lee, "An efficient key predistribution scheme for wireless sensor network using public key cryptography," in *Proceedings of the 6th International Conference on Networked Computing and Advanced Information Management (NCM '10)*, pp. 402–407, August 2010.

[37] T. D. Subash and C. Divya, "Novel key pre-distribution scheme in wireless sensor network," in *Proceedings of the International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT '11)*, pp. 959–963, March 2011.

[38] S. Mitra, R. Dutta, and S. Mukhopadhyay, "A hierarchical deterministic key pre-distribution for WSN using projective planes," in *Ad Hoc Networks*, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pp. 16–31, 2012.

[39] G. Nakos and D. Joyner, *Linear Algebra with Applications*, Brooks/Cole, Pacific Grove, Calif, USA, 1998.

[40] *Linear Algebra*, Birkhäuser, Boston, Mass, USA, 1997.

[41] J. Spencer, *The Strange Logic of Random Graphs*, Algorithms and Combinatorics, Vol. 22, Springer, Heidelberg, Germany, 2000.

[42] P. Erdős and A. Rényi, "On random graphs I," *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.

[43] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii*

International Conference on System Siences (HICSS '00), pp. 323–327, Maui, Hawaii, USA, January 2000.

[44] K. T. Kim, B. J. Lee, J. H. Choi, B. Y. Jung, and H. Y. Youn, "An energy efficient routing protocol in wireless sensor networks," in *Proceedings of the 12th IEEE International Conference on Computational Science and Engineering (CSE '09)*, pp. 132–139, August 2009.

[45] N. Xiong, M. Cao, A. V. Vasilakos, L. T. Yang, and F. Yang, "An energy-efficient scheme in next-generation sensor networks," *International Journal of Communication Systems*, vol. 23, no. 9-10, pp. 1189–1200, 2010.

[46] K. T. Kim, C. H. Lyu, S. S. Moon, and H. Y. Youn, "Tree-based clustering(TBC) for energy efficient wireless sensor networks," in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA '10)*, pp. 680–685, April 2010.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 765973, 9 pages
http://dx.doi.org/10.1155/2013/765973

Research Article

Real-Time Seismic Data Acquisition via a Paired Ripple Transmission Protocol

Jin-Ling Lin,[1] Kao-Shing Hwang,[2] Hui-Kai Su,[3] and Min-Che Hsieh[4]

[1] *Shih Hsin University, Taipei 11678, Taiwan*
[2] *National Sun Yat-sen University, Kaohsiung 80424, Taiwan*
[3] *National Formosa University, Yunlin 63201, Taiwan*
[4] *National Chung Cheng, Chiayi 62102, Taiwan*

Correspondence should be addressed to Jin-Ling Lin; jllin@cc.shu.edu.tw

Received 24 November 2012; Revised 4 April 2013; Accepted 23 April 2013

Academic Editor: Laurence T. Yang

This work uses a low-cost, reliable, and microchip-based wireless transmission solution to real-time collect earthquake data across local and wide areas. A transmission chain consisting of sensor units (nodes), each transmitting earthquake data unidirectionally to the end, is proposed. Each node consists of a seismic sensor, analog digital converter, radio frequency module, and a microchip for central control. The terminal node is responsible for transmitting data to a display server, which collects and analyzes all earthquake data from different transmission chains. Moreover, users also can distribute nodes, plug-in computers, in a wide area to monitor earthquake activities and transmit data to a web server. Then interested people can view the circumstance of an earthquake via web maps. For efficient wireless transmissions and to maximize bandwidth usage, a modified ripple protocol is applied to the wireless transmission between nodes in a daisy chain. Field experiments verify the practicality of the proposed system.

1. Introduction

Taiwan is located in an active earthquake zone on the western Pacific Ocean Rim. The convergence of the western Eurasian plate and the eastern Philippine Sea plate causes Taiwan's earthquakes. The danger to property and humans from earthquakes is serious. Thus, developing a seismic monitoring system, which can catch an earthquake's signals and analyze earthquake data is important [1–5].

Sensors are often used in harsh environments. Therefore, how to develop a reliable sensing solution for monitoring the variation of environment becomes very challenging [6]. A seismic monitoring system needs many sensors, sited at many locations, and all of these sensors must send signals to a collection center. The system consists of sensors, signal converters, data transmitters, data storage, and a user interface. First, sensors create signals using analog voltage. Next, signal converters change the analog signals to digital signals and then transmit the digital data to a data storage device. Finally, users can access the data and analyze it via the user interface. Data acquisition is the basic unit for the success of monitoring system [7]. An overview of a seismic monitoring system is shown in Figure 1.

When the system uses wired transmission, all sensors connect directly to the signal converter so the system is very reliable and can transmit a lot of data simultaneously. However, it is very inconvenient and time consuming to set up a wired transmission system. Wireless transmission overcomes the drawbacks of wired transmission [3–5, 9]. Besides, it is more practical as the system can observe very small signal changes with high resolution. So it would be better if this system could convert analog signals to high-resolution digital signals. A practical and reliable seismic monitoring system should be designed according to these requirements: wireless transmission, high-resolution analog-digital conversion, good reliability, convenience of accessing data, and low cost. Ripple transmission is reliable than the regular wireless transmission in such a seismic monitoring system, but it works on a computer instead of a microchip. Therefore, the article proposed the design of a cheaper RF

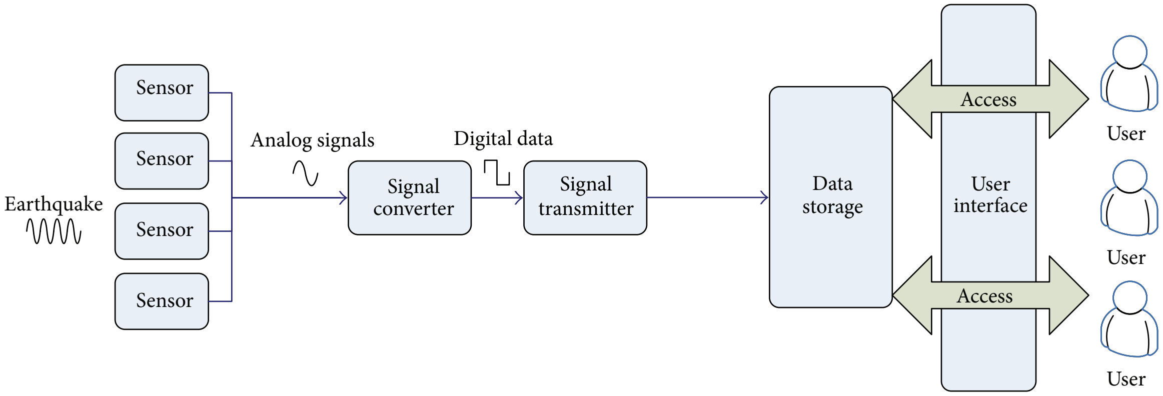


FIGURE 1: An overview of a seismic monitoring system.

module with a secure transmission protocol which is actually a simplified version evolving from the well-known ripple protocol.

The paper is organized as follows. Section 2 discusses background research into ripple wireless transmission protocol. Section 3 presents the architecture of the proposed seismic monitoring system. Section 4 discusses the research approach and system implementation. Section 5 presents the experiments and a discussion. Conclusions are provided in Section 6.

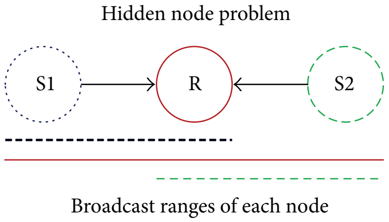


FIGURE 2: Hidden node problem.

2. Ripple Wireless Transmission Protocol

For wireless transmission, the most popular protocol [10] is carrier sense multiple access with collision avoidance (CSMA/CA). CSMA/CA is a protocol used to prevent data collisions. Once a node wishes to transmit data, it has to first listen to the channel for a predetermined amount of time to determine whether or not the channel is clear within the wireless range. If the channel is clear then the node is permitted to begin transmission; otherwise the node defers its transmission for a random period of machine cycle time. The use of collision avoidance is used to improve the performance of CSMA by attempting to divide the wireless channel up somewhat equally among all transmitting nodes within the collision domain. One of the problems of wireless data communications with CSMA/CA protocol is the hidden node problem. Figure 2 shows the hidden node problem, whereby node S1, in range of the receiver R, is not in the range of node S2; therefore, there is no way for S1 to know whether S2 is transmitting to R.

IEEE 802.11 RTS/CTS exchange supplements CSMA/CA by adding the exchange of a request to send (RTS) packet sent by the sender S and a clear to send (CTS) packet sent by the intended receiver R, such that all nodes within the range of the sender, receiver, or both are alerted not to transmit for the duration of the main transmission. Thus, the hidden node problem can be solved [11]. But, CSMA/CA with RTS/CTS exchanges cause another transmission problem—the exposed node problem, which occurs when a node is prevented from sending packets to other nodes due to a neighboring transmitter [12]. Suppose node S1 and node S2 want to send packages to node R1 and node R2, respectively, as shown in Figure 3. When node S1 sends an RTS packet to node R1, node R1 will send a CTS packet to node S1 and node S2, because S1 and S2 are in the transmission range of R1. In such a case, node S2 cannot send its data package to node R2 while node S1 is sending its data package to node R1. It is obvious that S2 can still send its data packages to R2 without interfering with the transmission between S1 and R1. The channel between S2 and R2 is wasted in such a case. Thus, a ripple protocol for wireless transmission is proposed to solve these transmission problems, within the CSMA/CA protocol.

Some wireless transmission protocols in the fashion of rippling [13, 14] is proposed to solve hidden node and exposed node problems, which arise when the CSMA/CA protocol is applied. In general, a ripple protocol with a chain topology uses a decentralized controlled-access approach to protect nodes from unintentional packet collisions, so it has the advantage of an ability to reuse space when compared with CSMA/CA protocol. Nodes are equally spaced, and radio nodes that are not neighbors do not interfere with each other in a ripple protocol. The ripple protocol has the advantages of identical ranges both in interference and transmission and a fixed elapsed time in data packet transmission.

A ripple protocol is proposed in this article which, as the protocol in [8], uses six types of frames: DATA, NULL, ready-to-send (RTS), clear-to-send (CTS), acknowledge (ACK), and ready-to-receive (RTR). The RTS and RTR frames are treated as "tokens," which is the same as other token-passing protocols. A node, in a ripple protocol, is allowed to send a DATA frame only if it holds a token. The way to generate

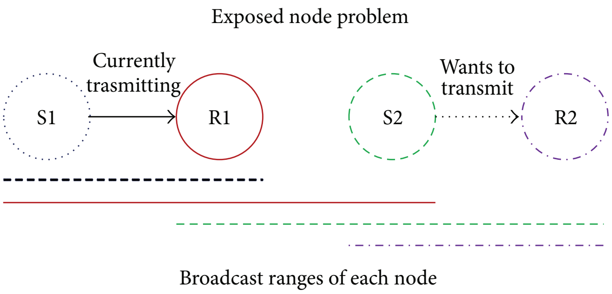


Figure 3: Exposed node problem.

and circulate RTS and RTR tokens depends on the state of the node. As shown in Figure 4, each node has four states.

(i) *Transmit (TX)*: a node will enter this state if it is ready to send a DATA frame.

(ii) *Receive (RX)*: a node will enter this state if it is ready to receive a DATA frame.

(iii) *Listen*: a node will enter this state if it overhears either CTS frame (for a hidden node) or RTS frame (for an exposed node). A node in Listen state must keep silent waiting for a network-allocation-vector (NAV) and may transmit an RTR token to its upstream node if the channel is sensed to be clear for IFS_{RTR} (the interframe space of an RTR frame) after the expiry of NAV.

(iv) *Idle*: it is the initial state for all nodes. A node will enter this state if it has been interrupted by unexpected conditions when in the TX, RX, or Listen state.

IFS_{RTR}, the interframe space (IFS) of an RTR frame, is set as two SIFS plus the time needed to transmit a RTS frame and defined as follows:

$$\text{IFS}_{\text{RTR}} = \text{SIFS} + T_{\text{RTS}} + \text{SIFS} = 2\text{IFS}_{\text{RTS}} + T_{\text{RTS}}, \quad (1)$$

where the IFSs of the remaining frames are all set as short IFS, SIFS. In the proposed system, IFS_{RTR} is treated as a one-time unit.

3. Architecture of the Seismic Monitoring System

The data type for wireless transmission must be digital, so seismic sensors, combined with analog digital converters (ADC) and wireless transmission, are the basic unit for the proposed seismic monitoring system, and the basic unit is treated as a node in the system. These nodes combined with a terminal node, used to collect data, form a transmission chain, as shown in Figure 5.

Each node in the transmission chain can receive analog signals, convert analog signals to digital signals, and transmit these digital signals to the next node. Each node not only receives digital data from its previous (upstream) node in the transmission chain but also transmits the data, including the data collected from itself and its previous node, to the next (downstream) node. The terminal node connects or is embedded into a computer, which analyzes the collected earthquake data and transmits it to a remote web server for further processing.

Considering the features of cost [3], data rate, and development difficulty, RF is preferred as the wireless transmission device module, instead of Bluetooth, Wi-Fi, or ZigBee. For the high-resolution digital signal, LTC2440 [15], which is a 24-bit high-speed differential delta sigma ADC with selectable speed/resolution and with an output rate up to 3.5 KHz, was chosen as the ADC converter. Finally, the microcontroller dsPIC33FJ128MC802 [16] was selected as the central controller of each node, instead of 8051 or ARM, when cost, signal processing ability, and development difficulty were taken into consideration.

4. System Implementation

One of the major objectives of this work is to alleviate the burden of carrying a bunch of communication cables to connect sensory nodes in deployment. IEEE 802.11 solution may be a good choice to resolve this concern. Whereas, while the cost and setup should be taken into considerations, this solution may turn out to be unfeasible, especially dealing with the tremendous number of deployed nodes. Therefore, the article proposed the design of a cheaper RF module with a secure transmission protocol which is actually a simplified version evolving from a well-known ripple protocol.

The integrity of data packages and transmission efficiency is the most important issue in wireless transmission. To maximize the usage of wireless transmission bandwidth and preventing packet collisions, the ripple protocol is ported on dsPIC8022 microchip and modified as a paired ripple transmission protocol (PRTP) such that it can appropriately work in the proposed system.

The paired ripple transmission (PRT) works on a microchip (dsPIC802s) instead of a computer, which the ripple transmission usually works on. The dedicated device mainly consisted of a 16 bits dsPic but capable of a reliable RF data transmission so as to being dispersed broadly in a field. The handshaking packet format of ripple transmission protocol is simplified and modified to 3 items: TYPE, NODE, and DES_NODE, and each item accounts for 1 byte only. TYPE stands for the type of packet, for example, RTS, CTS, RTR, or ACK. NODE and DES_NODE describe the source and destination of packet, respectively. Because the length of a DATA package is fixed, the network allocation vector (NAV) for each transmission is the same. Each node has to transmit on two different wireless channels, so the proposed paired ripple transmission needs two radio frequency (RF) modules—one is a receiving channel and the other is a sending channel. A node will communicate with its upstream node while the receiving channel and downstream node with the sending channel. The functions and processing flow of each node, which is ported with the proposed paired ripple transmission protocol, are described in Figure 6. Blue solid lines denote the possible input, and green dashed lines denote the output.

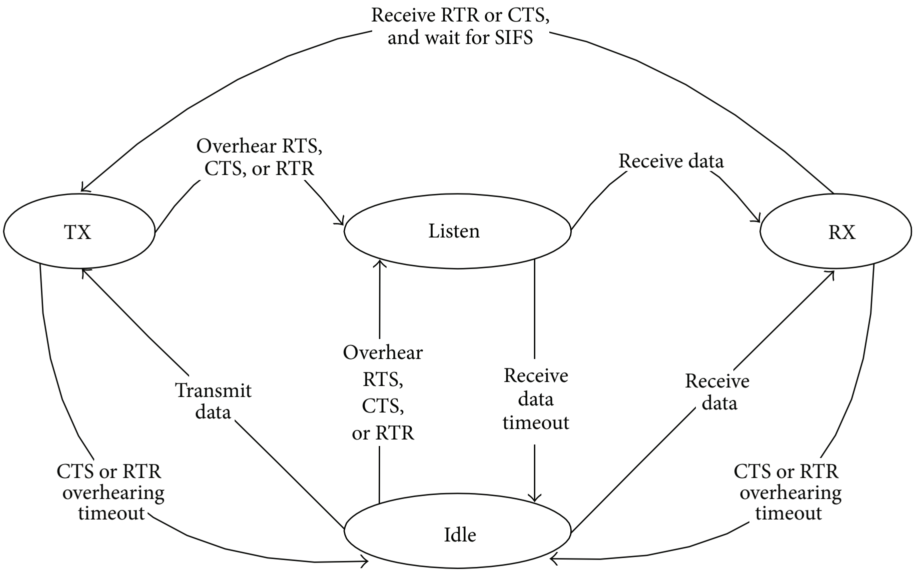


Figure 4: State diagram of a node in ripple protocol [8].

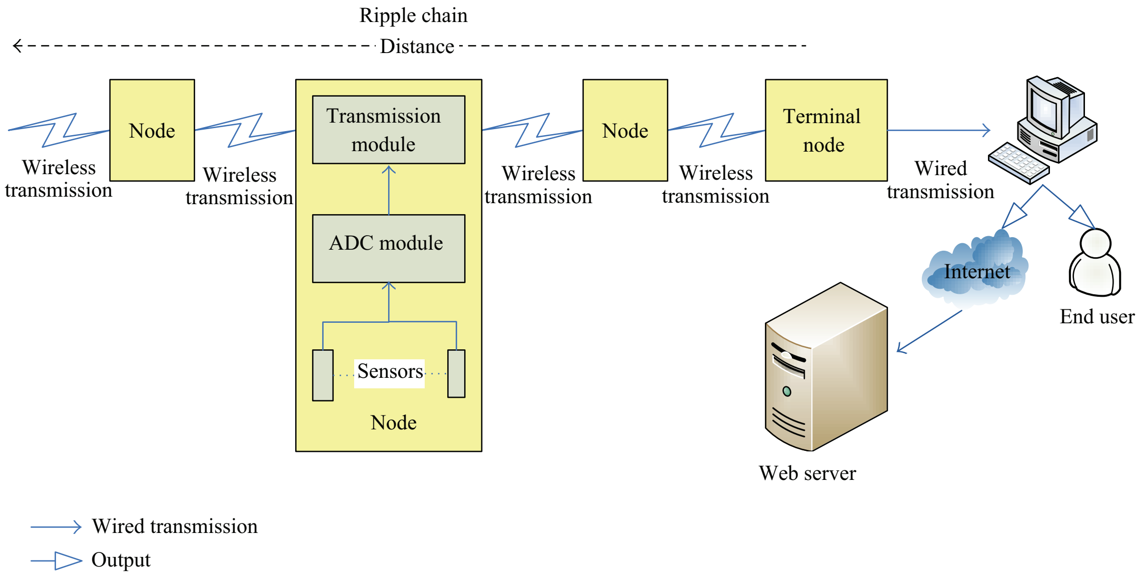


Figure 5: Structure of transmission in a daisy chain.

Figure 7 describes how a node changes its states with the paired ripple transmission protocol. Each node has two RF modules, one for receiving data packages from the upstream node, denoted by RX-RF, the other one for sending data packages to the downstream node, denoted by TX-RF. The background to the upper and lower part of an arrow represents the module RX-RF and TX-RF, respectively. When a node processes a handshake with other nodes, both of its two RF modules will enter TX mode to send packets to the upstream and downstream nodes. When a node listens to the channel or expects for a packet, both of its two RF modules are in RX mode. When a node is transmitting DATA frame, its RX-RF module will enter RX mode and the RF-TX module will enter TX mode.

The hardware needed in the proposed paired ripple transmission consists of a printed circuit board (PCB) and some electrical parts, shown in Figure 8. The dsPIC microcontroller takes charge of analog-digital conversion and RF transmission. Input data comes from the G-sensor and RF-RX. RS-232 and RF-TX are for output. Two RF modules are responsible for wireless transmission. If the node is the last one in the chain of transmission, data will be transmitted by RS-232 instead of RF. The power supply has two types of power sources: a 9 V Li-ion battery and a 5 V DC power jack. The 5 V power is supplied by a 5 V DC power jack or the output of a L7805, which is a positive voltage regulator and translates the voltage provided by the Li-ion battery from 9 V to 5 V. The 5 V power supply is used to drive a LTC2400,

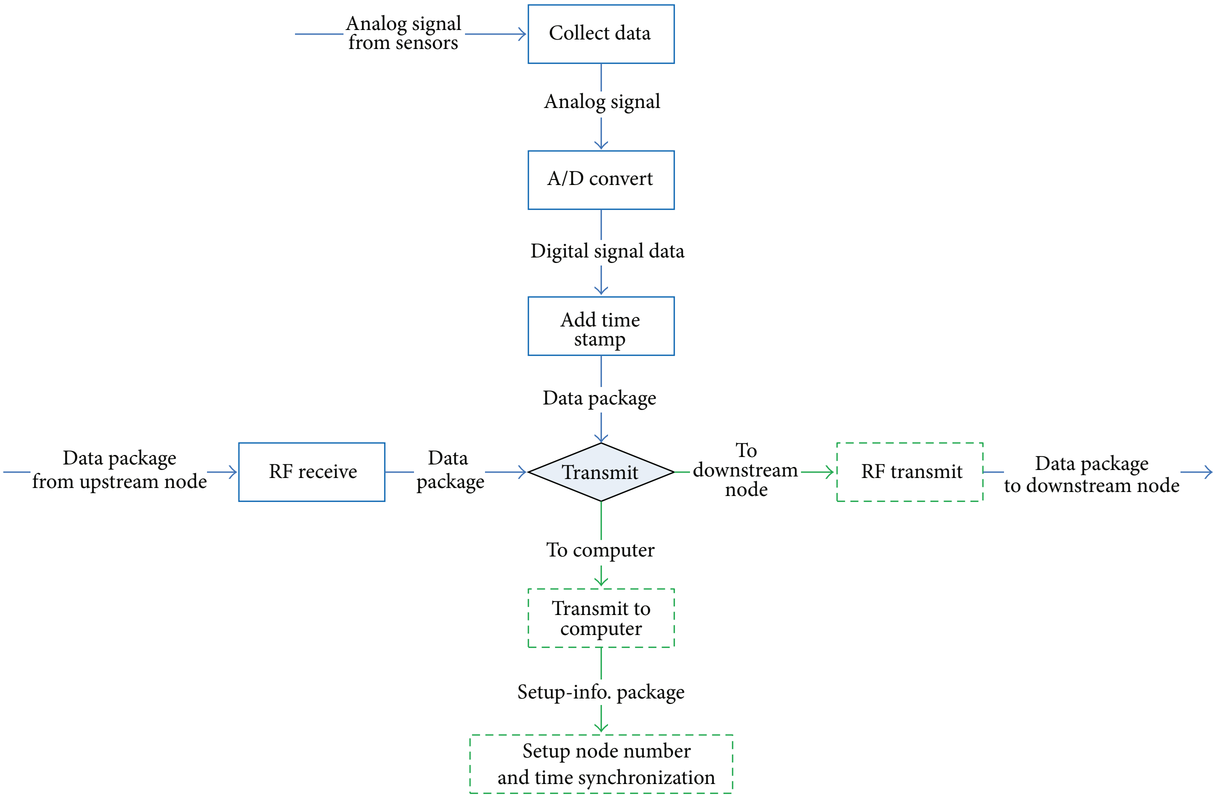


FIGURE 6: Functions and processing flow of each node in chained paired ripple transmission.

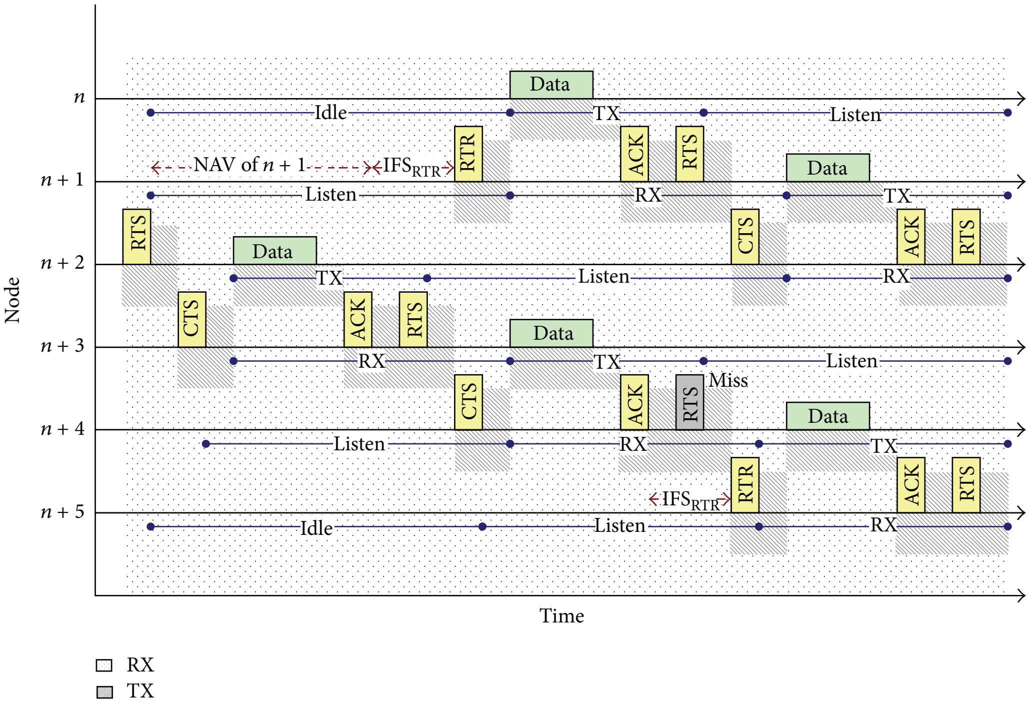


FIGURE 7: State diagram of RF modules in the proposed paired ripple transmission.

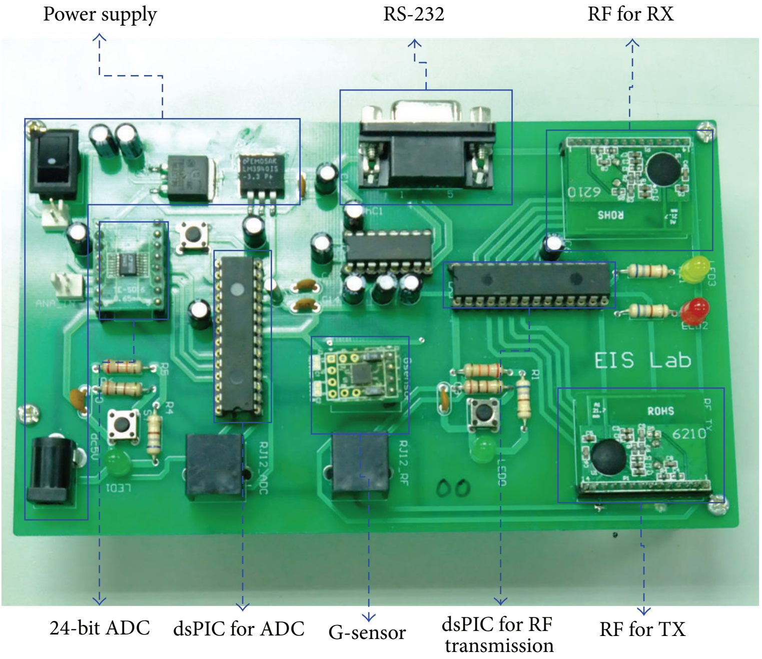


Figure 8: The printed circuit board for a node used in the proposed system.

G-sensor, and HIN232. The LM3940, in the power supply circuit, will reduce the 5 V down to 3.3 V and supply the dsPICs and RF modules.

Remote earthquake data acquisition and analysis need software tools to help with data acquisition, transmission, collection, and analysis. The software tools needed in the transmitting nodes, data-collection computer, and data-analysis server are described as following.

(1) Software Tools for the Transmitting Node. The software architecture in a node is divided into several parts: initial functions, ADC data framing functions, and RF communication functions, as shown in Figure 9. Initial functions are responsible for all of the hardware device initialization such as I/O mapping, communication interface, timer, and RF module initialization. The ADC data framing function is responsible for getting ADC value and combining it with a time stamp to be framed as an ADC data package. RF communication functions process data packages both from RF for RX and its own pair dsPIC802 for ADC. Then they transmit the input data package to the downstream node or computer.

(2) Software Architecture for the Data Collection Computer. The main mission of the data collection computer is transmitting data from the end node to the web server and doing time synchronization with the nodes. Users can set up the ID of each node and send time synchronization information at the same time. If the node is combined with the data collection computer, its ID is localhost, whose IP is 127.0.0.1; otherwise the ID can be any effective IP address. The data collection computer listens to COM port and receives messages continuously. Any message coming from COM port will be transmitted to the web server, by socket, immediately. In addition, this program can send a package which includes information about node setup and system time on the computer. Also, a visual component library (VCL), called TYbCommDevice, was added such that the program can support all COM ports.

(3) Software Architecture for the data Analysis Server. The main purpose of the server software is to make the APM work stably and build a friendly user interface. The program on the server can receive HTTP POST data packages from the socket and store data after analyzing the data package. The program can also respond to users' requests on the web browser, as shown Figure 10.

5. Experiments and Discussion

The dedicated device mainly consisted of a 16 bits dsPic but capable of a reliable RF data transmission so as to being dispersed broadly in a field. Therefore, to evaluate the performance of the proposed paired ripple transmission (PRT), the PRT was compared with the regular wireless transmission (RWT)—RF without using ripple transmission. The baud rate of the RS-232 is 115200, the length of each data package was 17 bytes, and the wireless transmission speed was 1 Mbps in the experiments. The numbers of data packages, transmitted and recorded on the server, were accounted for each minute.

First, the maximum efficacy of nodes in ripple protocol was tested. In Figure 11, the effective data, represented by a red line with square marker, means corrected data is recorded at the web server, the timeout missing data, represented by a

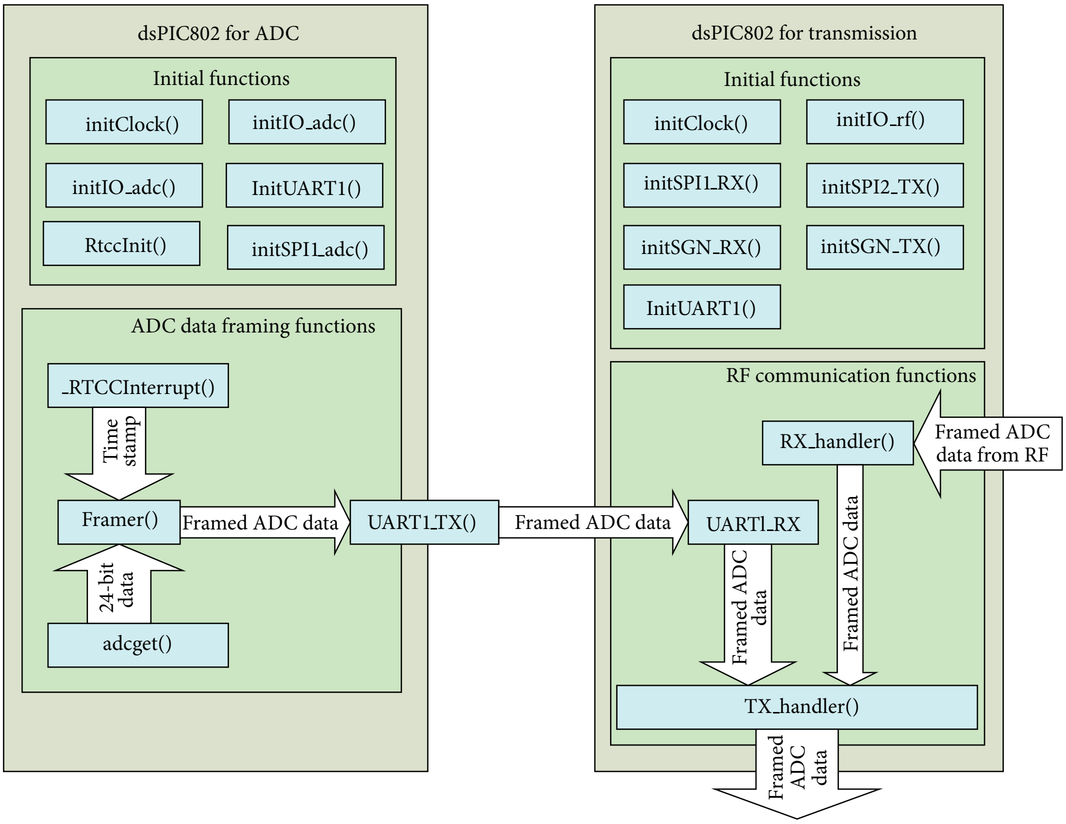
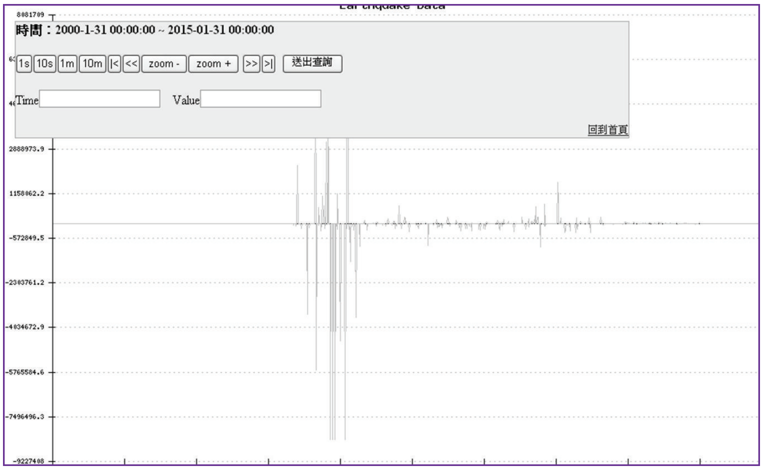


Figure 9: The software architecture and data flow for the transmitting node.


時間：2000-1-31 00:00:00 ~ 2015-01-31 00:00:00

Time

Value

Figure 10: An example of querying output from the data server.

green line with triangle marker, means these are remaining packages, which have not been sent to the web server, and the error data, represented by a black line with star marker, means the packages were garbled or in the wrong format. In Figure 11, someone can easily observe that the effective data is less than invalid data only when each node takes 125 samplings per second (SPS), since higher SPS would make the system too busy to process data. Besides, the error data increases when SPS becomes higher. In other words, the seismic monitoring system could work well when SPS is less than 125.

Figures 12 and 13 show the status of data packages for the proposed paired ripple transmission, which has 6 nodes in the chain of ripple transmissions and the regular wireless transmission, respectively. The brown brick denotes effective data, sky blue solid rectangles denote error data, and dotted

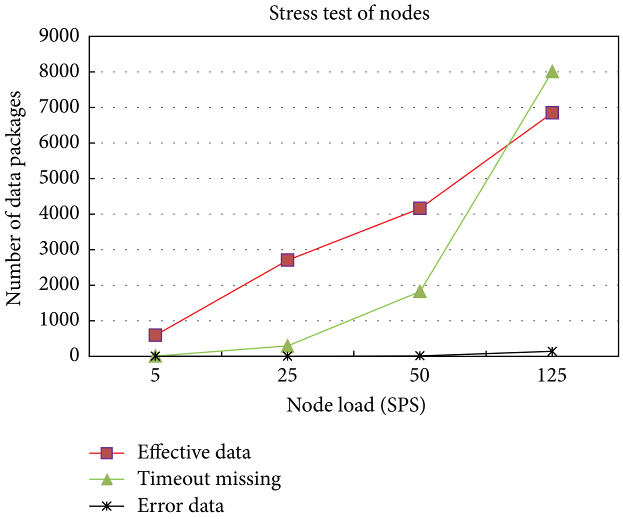


Figure 11: An example of the stress test of nodes (2 nodes only).

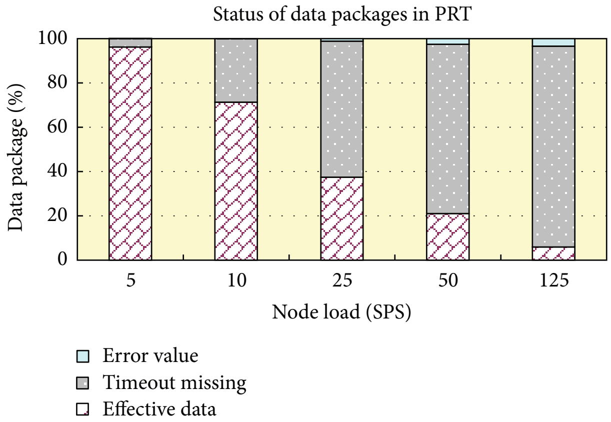


Figure 12: Status of data packages in the proposed paired ripple transmission (PRT).

gray rectangles denote timeouts-missing data. Since the regular transmission could result in the overrun of memory buffers and system crashes, the efficiency of transmission is affected. Ripple ensures an orderly transmission, but it still misses a lot of data when the data flow is over loading the transmission chain. From the experimental results, it is obvious that higher SPS leads to the less effective data both in paired ripple and regular transmissions. The regular transmission has much more error data than the proposed paired ripple transmission, as shown in Figures 12 and 13, and the number of effective data packages in paired ripple transmission is more than in regular transmissions, as shown in Figure 14, where the solid line represents the amount of effective data for PRT and the dashed line represents the amount of effective data for RWT.

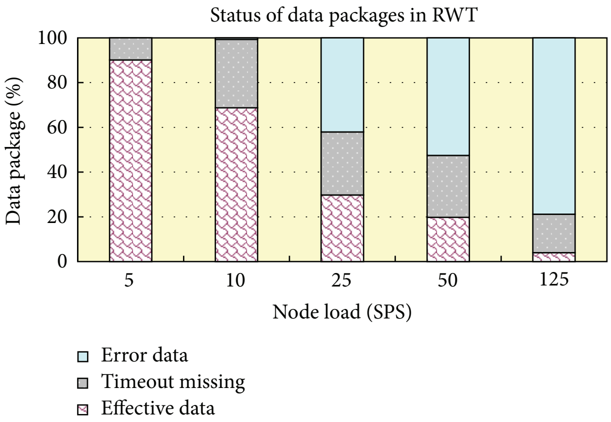


Figure 13: Status of data packages in the regular wireless transmission (RWT).

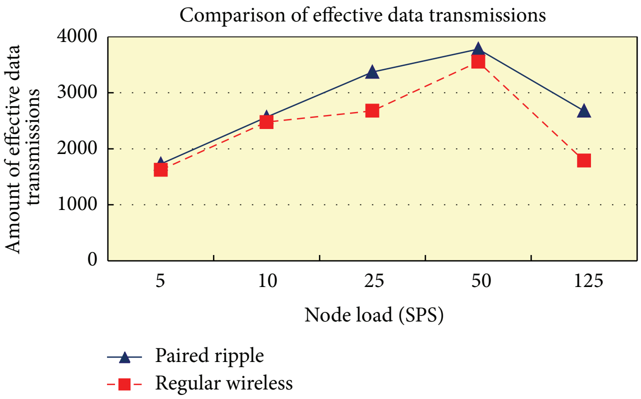


Figure 14: The comparison of the amount of effective data transmissions for RWT and PRT.

6. Conclusions

A paired ripple transmission protocol was proposed and implemented into modular hardware. The ripple protocol can improve transmission efficiency by spatial reuse in real-world applications. When the transmission chain gets overloaded, the regular wireless transmission has errors in large data packages, but the ripple transmission maintains the integrity of data packages. Because error data brings extra payloads to nodes, it makes the system crash. Thus, the proposed paired ripple transmission is more efficient than the wireless transmission without ripple. A wide area deployment is pictured, making the function of the designed work pieces more pervasive. A web server was set up for easy data representation, and a lot of cheap hardware parts were chosen, such that each node costs less than 30 dollars.

References

[1] Y. Pang and X. Liu, “The design of the intelligent and fieldbus sensor for measuringthe level and temperature of the water well monitoring eqrthquake,” in *Proceedings of the 41st SICE Annual Conference (SICE ’02)*, pp. 569–574, August 2002.

[2] T. C. Hutchinson and F. Kuester, "Monitoring global earthquake-induced demands using vision-based sensors," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 1, pp. 31–36, 2004.

[3] T. Takano and T. Maeda, "Experiment and theoretical study of earthquake detection capability by means of microwave passive sensors on a satellite," *IEEE Geoscience and Remote Sensing Letters*, vol. 6, no. 1, pp. 107–111, 2009.

[4] R. Tan, G. Xing, J. Chen, W. Z. Song, and R. Huang, "Quality-driven volcanic earthquake detection using wireless sensor networks," in *Proceedings of the 31st IEEE Real-Time Systems Symposium (RTSS '10)*, pp. 271–280, San Diego, Calif, USA, December 2010.

[5] J. Nachtigall and J. P. Redlich, "Wireless alarming and routing protocol for Earthquake Early Warning Systems," in *Proceedings of the 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS '11)*, Paris, France, February 2011.

[6] A. Ferreira Da Silva, A. F. Goncalves, L. A. De Almeida Ferreira et al., "A smart skin PVC foil based on FBG sensors for monitoring strain and temperature," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 7, pp. 2728–2735, 2011.

[7] P. Mariño, F. Poza, S. Otero, and M. A. Dominguez, "Reconfigurable industrial sensors for remote condition monitoring and modeling," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 4199–4208, 2010.

[8] R. G. Cheng, C. Y. Wang, and J. S. Yang, "Distributed medium access protocol for wireless mesh networks," U.S. Patent 7822009, http://www.freepatentsonline.com/7822009.html .

[9] R. C. Luo and O. Chen, "Mobile sensor node depolyment and asynchronous power management for wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 5, pp. 2377–2385, 2012.

[10] H. Liu, B. Zhang, H. T. Mouftah, X. Shen, and J. Ma, "Opportunistic routing for wireless ad hoc and sensor networks: present and future directions," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 103–109, 2009.

[11] H. Jasani and N. Alaraje, "Evaluating the performance of IEEE 802.11 network using RTS/CTS mechanism," in *Proceedings of the IEEE International Conference on Electro/Information Technology (EIT '07)*, pp. 616–621, Chicago, Ill, USA, May 2007.

[12] J. Jafarian and K. A. Hamdi, "Analysis of the exposed node problem in IEEE 802. 11 wireless networks," in *Proceedings of the 11th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, Livepool, UK, 2010.

[13] R. G. Cheng, C. Y. Wang, L. H. Liao, and J. S. Yang, "Ripple: a wireless token-passing protocol for multi-hop wireless mesh networks," *IEEE Communications Letters*, vol. 10, no. 2, pp. 123–125, 2006.

[14] R. G. Cheng, C. Y. Wang, and L. H. Liao, "Ripple: a distributed medium access protocol for multi-hop wireless mesh networks," in *Proceedings of the IEEE 63rd Vehicular Technology Conference (VTC '06)*, pp. 289–293, Melbourne VIC, Australia, July 2006.

[15] Linear Technology Inc, "LTC2440—24-bit high speed differential delta sigma ADC with selectable speed/resolution," 2012, http://www.linear.com/product/LTC2440 .

[16] Microchip Technology Inc., "16-bit dsPIC digital signal controllers—dsPIC33FJ128MC802," 2012, http://www.microchip.com/wwwproducts/devices.aspx?dDocName=en532302.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 108968, 17 pages
http://dx.doi.org/10.1155/2013/108968

Research Article

Efficient and Secure Routing Protocol Based on Encryption and Authentication for Wireless Sensor Networks

Jiliang Zhou

Shanghai University of International Business and Economics, Shanghai 201620, China

Correspondence should be addressed to Jiliang Zhou; hnzhoujl@126.com

Received 5 November 2012; Revised 10 March 2013; Accepted 26 March 2013

Academic Editor: Anfeng Liu

One concerned issue in the routing protocol for wireless sensor networks (WSNs) is how to provide with as much security to some special applications as possible. Another is how to make full use of the severely limited resource presented by WSNs. The existing routing protocols in the recent literatures focus either only on addressing security issues while expending much power or only on improving lifetime of network. None of them efficiently combine the above-mentioned two challenges to one integrated solutions. In this paper, we propose efficient and secure routing protocol based on encryption and authentication for WSNs: BEARP, which consists of three phases: neighbor discovery phase, routing discovery phase, and routing maintenance phase. BEARP encrypts all communication packets and authenticates the source nodes and the base station (BS), and it ensures the four security features including routing information confidentiality, authentication, integrity, and freshness. Furthermore, we still design routing path selection system, intrusion detection system, and the multiple-threaded process mechanism for BEARP. Thus, all the secure mechanisms are united together to effectively resist some typical attacks including selective forwarding attack, wormhole attacks, sinkhole attacks, and even a node captured. Our BEARP especially mitigates the loads of sensor nodes by transferring routing related tasks to BS, which not only maintains network wide energy equivalence and prolongs network lifetime but also improves our security mechanism performed uniquely by the secure BS. Simulation results show a favorable increase in performance for BEARP when compared with directed diffusion protocol and secure directed diffusion protocol in the presence of compromised nodes.

1. Introduction

A wireless sensor network (WSN) is a collection of nodes that can form a network without the need of a fixed infrastructure, which operates in an unattended, sometimes hostile, environment. Nodes can be connected arbitrarily, and all nodes take part in discovery and maintenance of routes to other nodes in the network [1]. Thus, one concerned issue when designing wireless sensor network is the routing protocol that requires the researchers to provide as much security to the application as possible [2]. Another important factor makes full use of the severely limited resource presented by WSNs, especially the energy limitation. Current presenters for routing protocols in sensor networks optimize for the limited capabilities of the nodes and the application specific nature of the networks, or they incorporate security into these proposed protocols; however, they have not been designed with security as a goal. When the defender has the liabilities of insecure wireless communication, limited node capabilities, and possible threats, and the adversaries can use powerful laptops with high energy and long-range communication to attack the network, therefore, designing a secure routing protocol for WSNs is crucial and nontrivial [3, 4].

1.1. Background. There are two secure problems to be considered when designing a secure routing protocol. On the one hand, different from the special router between conventional networks connected by wire cable, any node in sensor networks can be a router which can not only route to another node but also receive and send any routing information in a certain scope. On the other hand, one aspect of sensor networks that complicates the design of a secure routing protocol is in-network aggregation and inside attacks. In more conventional networks, a secure routing protocol is typically only required to guarantee message availability. Message confidentiality, authenticity, integrity, and freshness are handled at a higher layer by an end-to-end security mechanism. End-to-end security is possible in more conventional networks because it is absolutely unnecessary for intermediate routers to have access to the content of messages [5, 6]. However, in sensor networks, in-network processing makes end-to-end security mechanisms harder to deploy because intermediate nodes need direct access to the content of the messages. Link layer security mechanisms can help mediate some of the resulting vulnerabilities, but it is not enough for WSNs: we will now require much more from conventional routing protocols.

1.2. Related Works. In this section, we will discuss directed diffusion (DD) protocol and secure DD protocol (S-DD), possible attacks on routing protocol, securing routing protocols, and detecting compromised nodes. DD protocol and secure DD protocol, as our research and comparison representative, are very important, even a milestone, for the research of routing protocol for WSNs, and above all secure routing must exert to aim at some kinds of possible attacks. Moreover, it is necessary for the whole security mechanism to be improved by detecting the compromised nodes in case the routing protocol fails. The following is the current related work for them.

1.2.1. DD Protocol and S-DD Protocol. DD protocol [7] consists of several elements: interests, data messages, gradients, and reinforcements. An interest message is a query or an interrogation which specifies what a user wants. Each interest contains a description of a sensing task that is supported by a sensor network for acquiring data. In general speaking, in DD protocol, the setup and maintenance of extensive routing table are avoided. Instead, it relies on the broadcast propagation of queries, pruned by information content and geographical data. Sensor nodes maintain route caches which contain the source routes of the other nodes that are known. All in all, DD protocol is not resource aware or resource adaptive and especially suffers from many attacks for lack of encryption and authentication in course of packet receiving and transmitting.

Then, Wang et al. [8] present the design of a new secure directed diffusion protocol (S-DD), which provides a secure extension for the directed diffusion protocol. They mainly focus on secure routing and give a simple scheme to securely diffuse data, which uses an efficient one-way chain and do not use asymmetric cryptographic operations in this protocol. However, S-DD cannot work against any active attackers or compromised nodes in the network with the in-network aggregation. Especially, all sensor nodes do not have the ability to authenticate their neighbor nodes, so S-DD is also not robust without the in-network aggregation.

1.2.2. Possible Attacks on Routing Protocol. Two kinds of attacks can target routing protocols for WSNs [9]: passive attacks, where the attacker just eavesdrops on the routing information, and active attacks, where the attacker impersonates other nodes, drops packets, modifies packets, launches denial of service attacks, and so forth.

Most of the current routing protocols assume that all nodes in the network are trustworthy. The control information in the header of the packets carries the routing information, and intermediate nodes are assumed not to change this information. However, a compromised node can easily change the routing field of the packet and redirect the packet to anywhere it wants. The attacker can also redirect the route by changing the route sequence number in some protocols. In that case, the attacker can divert the traffic to itself by advertising a route to a node with the base station (BS) sequence number which is greater than the BS node's route. Redirecting the traffic can also be established by modifying the hop count. Route length is represented as hop count in routing protocol. A compromised node can direct all the traffic to itself by broadcasting the shortest hop count.

These attacks include impersonation, fabrication, and wormhole attacks. Compromised nodes can also create loops by changing the routes in the data packets. This will result in denial of service attacks. In impersonation, the attacker pretends to be another node after learning its address and changes it to its own address. Fabrication is another attack, where the compromised node generates false route messages, such as false error messages. In DD protocol, when links go down and routes break, the node which precedes this broken link broadcasts a "route error message." A compromised node can easily send false error messages for a working route. Another attack is the route cache poisoning attack. Any node can overhear the traffic, and if it finds route information, it adds it to its cache for future use. A compromised node can then broadcast spoofed packets with source route via itself. Then, neighboring nodes hear this and add the route to their cache. Also a compromised node can attack the routing table by overflowing it. It can attempt to initiate route discovery to nonexisting nodes. The worst attack is node captured, in which all information may be exposed and decrypted.

Finally, multihop routing in WSNs causes the packets to be delivered between one or more intermediate nodes. The security of routing information is harder to manage in this case.

1.2.3. Secure Routing Protocols for WSNs. Secure routing protocols for WSNs are difficult to be designed, especially when the nodes of a wireless sensor network have limited resources such as low battery power, CPU processing capacity, and memory. Since most routing protocols currently assume that nodes are trustworthy, security in WSNs mainly deals with authentication of the user nodes and security of the data packets that are being routed. Authentication is one goal, which verifies the identity of a node. A BS, a key, or the use of certificates can be implemented to perform authentications. Certificates can be thought of as a unique identification for every node. In Internet of Things, security mechanisms based on access control and secret communication channels regarding defending against outside attackers have been studied [10].

Zhou and Haas [11] proposed the idea of distributing a BS throughout the network in a threshold fashion. However, Zhou and Haas adopted public key and threshold cryptography, which are very expensive for sensor devices. Therefore, we do not consider this method practical for the time being. All the protocols below assume the preexistence and presharing of secret keys for all honest nodes in the beginning.

Adrian Perrig and Robert Szewczyk [3] present a suite of security protocols optimized for sensor networks: SPINS [5]. SPINS has two secure building blocks: SNEP and μTESLA. SNEP includes data confidentiality, two-party data authentication, and evidence of data freshness. μTESLA provides authenticated broadcast for severely resource-constrained environments. However, their system requires synchronized clocks for all the nodes in the network, and the SPINS is not robust for routing attacker because it is not based on the secure routing protocol.

Secure routing protocol (SRP) proposed by Papadimitratos and Haas guarantees correct route discovery [12]. They assume a security association between the end points in the beginning. The correctness of their protocol was only proven analytically.

Nasser and Chen proposed an efficient routing protocol, which we called SEEM [13], for WSNs. Compared to other proposed routing protocols, SEEM is designed based on utilizing multipath concept and considers energy efficiency and security simultaneously. However, SEEM is not really secure because its packets can be modified by attackers without any encryption and authentication.

Lee and Choi have presented SeRINS [14]: a secure alternate path routing in sensor networks. Their alternate path scheme makes the routing protocol resilient in the presence of compromised nodes that launch selective forwarding attacks. It also detects and isolates the compromised nodes, which try to inject inconsistent routing information, from the network by neighbor report system. In neighbor report system, a node's route advertisement is verified by its surrounding neighbor nodes so that the suspect node is reported to the BS and is excluded from the network. We think the SeRINS has not combined the authentication with encryption, and cooperation of several neighbor nodes can make the reported information good in order to cheat the BS, so the packet of the verified itself is not secure, and this leads the whole protocol not to be secure and trusted.

1.2.4. Detecting Compromised Nodes. Compromised nodes in WSNs usually promise to forward packets but later drop the data packets and refuse to forward them. Current network protocols do not have a mechanism to detect such nodes. Link layer acknowledgment such as IEEE 802.11 MAC protocol can detect link layer failure. However, it cannot detect a forwarding failure. Some protocol acknowledgments can detect end-to-end communication failure, but it cannot detect which particular node caused the failure in between [15].

Some researchers propose the idea of having neighbor nodes detect each other's behaviors and then report to each other or to a network authority, which detects compromised nodes by observing the reports on several attacks in the network [14, 16, 17]. All nodes have a monitor and reputation records, trust records, and a path manager. All these adapt to changes in networks and find out the misbehaving nodes in the network. However, we think that compromised nodes acting in groups can make these records good for themselves without the authentication mechanism with encryption. Therefore, we believe that a special agent such as BS is necessary for intrusion detection system (IDS) to detect the compromised nodes.

1.3. Contributions. In this paper, we propose a new routing protocol BEARP: efficient and secure routing protocol based on encryption and authentication for WSNs. In BEARP, we design to encrypt all communication packets, authenticate the source node and the BS, and ensure the four security features including routing information confidentiality, authentication, integrity, and freshness. Moreover, BEARP mitigates the load of sensor nodes by transferring routing-related tasks to the BS which operates routing paths selection and intrusion detection system. In routing paths selection system, the BS periodically selects a newly best path from many paths based on current energy level of nodes along each path. In the process of selecting route, especially, we design the algorithm *multi_shortest_path* to create another child thread, which executes the function *send_rout*e in time when finding a route to the source node. This thread helps decrease the delay for sending routing information. In intrusion detection system, detecting compromised nodes also performs uniquely by the secure BS. Therefore, the two approaches not only maintain network wide energy equivalence and prolong network lifetime but also improve our security mechanism. BEARP can effectively resist to some typical attacks including selective forwarding attack, wormhole attacks, sinkhole attacks, and even a node captured.

Compared to other proposed routing protocols, BEARP not only considers integration between energy efficiency and security simultaneously but regards security as our design goal for the first time. At the same time, the feature making BEARP distinct is that BEARP takes full advantage of the predominance of the BS. As a result, packet delivery ratios and network lifetime for operating BEARP in the WSN are more preferable and work better against some attacks, compared to operating DD protocol. The contributions of our work include the following: (1) we implement the four security features for WSNs including routing information confidentiality, authentication, integrity, and freshness, and BEARP works well under some typical attacks; (2) BEARP has much better packet delivery ratio than DD protocol in the presence of some compromised nodes; (3) the network lifetime is prolonged compared to insecure routing protocols, like DD protocol; (4) BEARP has almost no blocked nodes in WSNs and remarkably surpasses DD protocol.

1.4. Organization of the Paper. Foregoing contents are our preliminary work before we propose BEARP. The following in this paper is organized as follows. Some used notations and assumptions are introduced in Section 2. In Section 3, we present BEARP routing protocol and related algorithm and give some implementation details. Then, we discuss the security analysis for BEARP in Section 4, followed by performance evaluation in Section 5. Finally, we draw our conclusions in Section 6.

2. Notations and Assumptions

Sensor networks typically consist of one or multiple base stations and hundreds or thousands of inexpensive, small, and hardware-constrained nodes scattered over a wide area. Our sensor network model includes a powerful BS and numerous constrained sensor nodes. BS, which has greater capabilities, can directly transmit data to any node in the network. Resource-constrained sensor node, whose transmission range is limited, can send data along the multihop route to the BS. We consider that a BS is trustworthy, differently from sensor nodes. Moreover, we can extend naturally our scheme for a single BS to multiple BS as presented by Deng et al. [18].

Developing a proper threat model against our routing protocols, we consider two attack sources: outer or insider [19]. Outsider attackers do not have trusted keys. They typically rely on message replay or delay to influence routing protocols. Insider threats occur when a fully trusted node, with appropriate key material, is compromised. We assume the key management system is always secure, since there have been a lot of successful researches for them. The attacks launched from outsiders cannot join in the network because of the assumption, but we consider that the outsider attackers can interfuse in the network to be compromised nodes through any other special means.

Before presenting our BEARP protocol, we introduce some used notations and assumptions about sensor network in Tables 1 and 2, respectively, which are used in the following sections.

3. Routing Protocol Based on Encryption and Authentication (BEARP)

Now we present our BEARP protocol, which consists of three phases: neighbor discovery phase, routing discovery phase, and routing maintenance phase. In the following, each of them will be described in detail [13, 20, 21].

3.1. Neighbor Discovery Phase. Neighbor discovery takes place right after the deployment of all sensor nodes. However, neighbor discovery can be launched at any time by the BS during the lifetime of the sensor network. By doing this, the BS can request to reconstruct this network topology according to the great changes of the topology [13, 20].

To initiate the neighbor discovery, the BS selects broadcast key *BK* to encrypt the packet neighbor discovery (*ND*) and broadcasts the packet confidential *ND* (*CND*) to the whole network. After receiving this packet, each node does as follows (see Table 4):

(1) decrypt $ND = D_{BK}(CND)$ with the broadcast key *BK* of the node;

(2) record the address *prev_hop* from which the current node receives the packet and stores it in the list *neighbor_list* in ascending order of packet received time;

(3) change the address *prev_hop* to the address of itself;

(4) check if the broadcast packet has been received by searching *pkt_seq_num* in the table *rc_pkt_table*. If the packet has already been received once, the node drops this *CND* and does not rebroadcast it. Otherwise, it stores *pkt_seq_num* in table *rc_pkt_table*, encrypts $CND = E_{BK}(ND)$ with the broadcast key *BK,* and rebroadcasts the *CND* to its neighbor.

The fourth step insures that no *CND* packet is broadcasted more than one time for each node, which also applies to other control messages. Thus, the communication overheads for transmitting control packets are reduced to a low level.

Through the process of receiving, decrypting, segmenting, encrypting, and rebroadcasting *CND*, each node knows its real neighbor and stores them for using in the following phases.

The BS waits for a short time to ensure that the *CND* broadcast can be flooded through the network. Then, the BS broadcasts another packet confidential neighbor collection (*CNC*) in order to collect the neighbor information of each node. At the same time, the BS sets the current time T_B and the random number R_B to the packet *NC* in order to authenticate each node in the WSN. After receiving this packet, each node does as follows (see Table 5(a)):

(1) decrypt $NC = D_{BK}(CNC)$ with the broadcast key *BK* of the node and gets the two fields T_B and R_B for creating the reply packet *CNCR*;

(2) check if the address *prev_hop* from which the current node receives the packet has been saved in the list *neighbor_list*. If not then it stores it;

(3) change the address *prev_hop* to the address of itself;

(4) check if the broadcast packet has been received by searching *pkt_seq_num* in the table *rc_pkt_table*. If the packet has already been received once, the node drops this *CNC* and does not rebroadcast it. Otherwise, it stores *pkt_seq_num* in table *rc_pkt_table*, keeps the other fields in the packet, encrypts $CNC = E_{BK}(NC)$ with the broadcast key *BK,* and rebroadcasts the *CNC* to its neighbor.

When sensor node receives the *CNC* packet, it replies a confidential neighbor collection reply (*CNCR*) packet to the BS by flooding. In *NCR*, we add the session key field *SK*, time field T_B, and random number field R_B, and the source address is set to itself, and the destination address is set to the BS. The *CNCR* packet contains the following information:

(1) the address of the node,

(2) the list that has all addresses of its neighbors,

(3) the session key between the node and the BS,

(4) the authentication information.

Table 1: Basic notations.

BK	The initial key used to create the session key between BS and nodes and encrypt the routing message at beginning.
SK	Session key used for data encryption and authentication between BS and source.
T_B, T_S	Denote the current time of the BS and the current time of the source node, respectively.
R_B, R_S	Denote the random number of the BS selected and the random number of the source node selected, respectively.
$E_k(x)$	Encryption of message x with key k.
$D_k(x)$	Decryption of cipher message x with the key k.
$x\|y$	Concatenation of message x and y.
prev_hop	Denote the previous node address from which the current node receives the packet
next_hop	Denote the next node address to which the current node sends the packet.
neighbors_list	Neighbors address list.
pkt_seq_num	The sequent number of a packet.
rc_pkt_table	Received packet table that stores the sequent number of packets.
route_list	The routing list field in a packet.
route_table	The routing table in a node.
pkt_type	Packet type including *CND*, *CNC*, *CNCR*, *CDE,* and *CDER*.
source_add	The source node address.
data_length	The length of data packet.
itself_add	The current node address.
$A \xrightarrow{M} B$	Node A sends message M to node B.

Table 2: Assumptions.

A-1:	The links between these sensor nodes are always bidirectional. The communication patterns in WSNs fall into three categories: node to BS, BS to node, and BS to all nodes.
A-2:	The BS has sufficient battery power to surpass the lifetime of all sensor nodes and sufficient memory to store cryptographic keys, and it is very secure and cannot be compromised under any conditions.
A-3:	Each node in WSNs has unique identifier stored in BS, and it can forward a message towards the BS, recognize packets addressed to it, and handle message broadcasts.
A-4:	WSNs may be deployed in unauthentic locations, and basic wireless communication is not secure. Individual sensors are untrustworthy; any adversary can eavesdrop on traffic, inject wrong routing messages, and replay old routing messages.
A-5:	Each node can get a master secret key which it shares with the BS before its deployment. The secret key is used as authentication key by the BS.
A-6:	The BS can update all secret keys between any nodes after a certain period of time, and the key management system is always secure.

Table 3: Neighborhood matrix.

	BS	1	2	3	4	5	6
BS	0	∞	∞	0	0	0	0
1	1000	0	1000	1000	1000	0	0
2	1000	1000	0	1000	1000	0	0
3	0	1000	1000	0	1000	1000	1000
4	0	1000	1000	1000	0	1000	1000
5	0	0	0	1000	1000	0	1000
6	0	0	0	1000	1000	1000	0

Each node receiving this packet does as follows (see Table 5(b)):

(1) decrypt $NCR = D_{BK}(CNCR)$ with the broadcast key BK of the node;

(2) check if the address *prev_hop* from which the current node receives the packet has been saved in the list *neighbor_list*. If not then it stores it;

(3) change the address *prev_hop* to the address of itself;

(4) check if the broadcast packet has been received by searching *pkt_seq_num* in the table *rc_pkt_table*. If the packet has already been received once, the node drops this *CNCR* and does not rebroadcast it. Otherwise, it stores *pkt_seq_num* in table *rc_pkt_table*, keeps the other fields in the packet, encrypts $CNCR = E_{BK}(NCR)$ with the broadcast key BK, and rebroadcasts the *CNCR* to its neighbor.

When the BS receives the packet *CNCR*, at first, it must authenticate communication time cost by comparing the time field T_B with the current time and authenticate the freshness of the packet by comparing random number field R_B with the foregone random number R_B. If the authentication fails, the BS will drop the packet. Finally, after receiving neighbor information of all nodes, the BS has a vision of the topology of the whole networks.

To select a path that has the maximum available energy on each node, we introduce the concept weight. The weight of an edge in the corresponding graph of the network is

Table 4: Confidential neighbor discovery packet broadcasts.

BS (sender)		Neighbor of BS (receiver)
$ND = [pkt\_type \Vert BS \Vert prev\_hop \Vert pkt\_seq\_num]$; $CND = E_{BK}(ND)$.	$\xrightarrow{\text{CND}}$	$ND = D_{BK}(CND); prev\_hop \rightarrow neighbor\_list;$ $itself\_address \rightarrow prev\_hop;$ $pkt\_seq\_num \rightarrow rc\_pkt\_table;$ $ND = [pkt\_type \Vert BS \Vert prev\_hop \Vert pkt\_seq\_num];$ $CND = E_{BK}(ND).$
N_i (sender)		N_{i+1} (receiver, neighbor of node N_i)
$ND = D_{BK}(CND); prev\_hop \rightarrow neighbor\_list;$ $itself\_address \rightarrow prev\_hop;$ $pkt\_seq\_num \rightarrow rc\_pkt\_table;$ $ND = [pkt\_type \Vert BS \Vert prev\_hop \Vert pkt\_seq\_num];$ $CND = E_{BK}(ND).$	$\xrightarrow{\text{CND}}$	$ND = D_{BK}(CND); prev\_hop \rightarrow neighbor\_list;$ $itself\_address \rightarrow prev\_hop;$ $pkt\_seq\_num \rightarrow rc\_pkt\_table;$ $ND = [pkt\_type \Vert BS \Vert prev\_hop \Vert pkt\_seq\_num];$ $CND = E_{BK}(ND).$

Table 5: Confidential neighbor collection and confidential neighbor collection reply packet broadcasts.

(a) Confidential neighbor collection packet broadcasts

BS (sender)		N_i (receiver)
$NC = [pkt\_type \Vert$ BS$\Vert prev_hop$ $\Vert pkt\_seq\_num \Vert T_B \Vert R_B]$; $CNC = E_{BK}(NC)$.	$\xrightarrow{\text{CNC}}$	$NC = D_{BK}(CNC);$ get T_B, R_B; $prev\_hop \rightarrow neighbor\_list$; $itself\_address \rightarrow prev\_hop$; $pkt\_seq\_num \rightarrow rc\_pkt\_table$; $NC = [pkt\_type \Vert BS \Vert prev\_hop \Vert pkt\_seq\_num \Vert T_B \Vert R_B]$; $CNC = E_{BK}(NC)$.

(b) Confidential neighbor collection reply packet broadcasts

Source node (sender)		N_i (receiver, neighbor of BS)
$NC = D_{BK}(CNC)$; $NCR = [pkt\_type \Vert$ source_add$\Vert neighbor_list$ $\Vert prev\_hop \Vert pkt\_seq\_num \Vert SK \Vert T_B \Vert R_B - 1]$; $CNCR = E_{BK}(NCR)$.	$\xrightarrow{\text{CNCR}}$	$NCR = D_{BK}(CNCR)$; $prev\_hop \rightarrow neighbor\_list$; $itself\_address \rightarrow prev\_hop$; $pkt\_seq\_num \rightarrow rc\_pkt\_table$; $NCR = [source\_add \Vert neighbor\_list \Vert prev\_hop$ $\Vert pkt\_seq\_num \Vert SK \Vert T_B \Vert R_B - 1]$; $CNCR = E_{BK}(NCR)$.
N_i (sender)		BS (receiver)
$NCR = D_{BK}(CNCR)\ (prev\_hop) \rightarrow neighbor\_list$; $itself\_address \rightarrow prev\_hop$; $pkt\_seq\_num \rightarrow rc\_pkt\_table$ $NCR = [source\_add \Vert neighbor\_list \Vert prev\_hop \Vert$ $pkt\_seq\_num \Vert SK \Vert T_B \Vert R_B - 1]$; $CNCR = E_{BK}(NCR)$.	$\xrightarrow{\text{CNCR}}$	$NCR = D_{BK}(CNCR)$; $pkt\_seq\_num \rightarrow rc\_pkt\_table$; get $source\_add$, $neighbor\_list$, SK, $R_B - 1$; authenticate T_B, $R_B - 1$.

the available energy on the head node. The BS then constructs a directed graph marked weight with neighbor information. The weight decreases as the head node sends and receives packets. Figure 1 shows the subgraph derived from the network topology. In Figure 1, the weights of edges starting from BS are infinite, which means that the BS has much more energy than other sensor nodes.

The calculation of the weight is based on the formula

$$\text{Weight} = \frac{\text{total power of each node}}{\text{power for transmitting or receiving one packet}}. \quad (1)$$

We assume the total energy of each node initially is 10000 units and the total energy for sending one packet is about 10 units; then

$$\text{weight} = \frac{10000}{10} = 1000 \text{ units}. \quad (2)$$

We use neighborhood matrix to represent the neighborhood relations between nodes. Table 3 shows the weighted matrix corresponding to the graph in Figure 1. Each row except the first row contains the neighbor information of a specific node; for example, the second row shows neighbor information of the BS. Each column except the first column represents a node. If the value for some space is not zero, it means that the nodes corresponding to the row and the

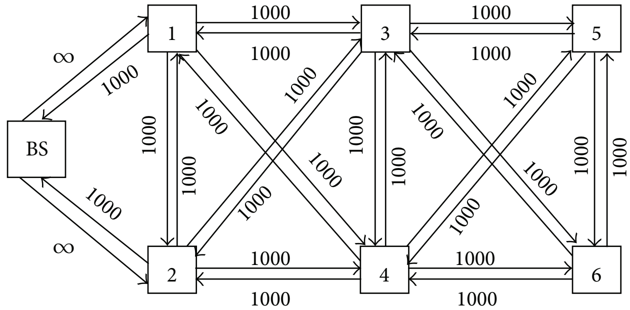


Figure 1: The subgraph derived from network topology.

column are neighbors. The value of each space is the weight of the edge from the node corresponding to the row to the node corresponding to the column. As we defined, weights of edges from the BS are infinite. For example, from Table 3 we know that *node 3* has five neighbors: *node 1, node 2, node 4, node 5,* and *node 6.*

3.2. Routing Discovery Phase. The BS starts its task, routing discovery, beginning at phase two. The task is divided into three subtasks: data enquiry; routing path selection system (RPSS); sending routing information.

3.2.1. Data Enquiry. By now, BEARP supports only data transmission requested by the BS; that is, the BS broadcasts enquiry for data with specific features. Sensor nodes have satisfied the enquiry response with enquiry reply. Data transmission follows these steps.

(1) The BS broadcasts an enquiry packet confidential data enquiry (*CDE*).

(2) Sensor nodes have satisfied the enquiry response with a reply packet confidential data enquiry reply (*CDER*).

(3) Sensor nodes that do not satisfy the enquiry rebroadcast *CDE.*

(4) The BS calculates a shortest path to the desired node in the weighted graph. The shortest path is a path from the source to the BS of which the total energy consumed on each node for sending one packet is the least, that is, usually the path with minimum hops.

Each node receiving *CDE* packet does as follows:

(1) check if it satisfies the enquiry itself;

(2) if not, the node rebroadcasts the *CDE* and saves the *pkt_seq_num* to avoid repeating broadcasting the *CDE* more than once;

(3) if it does, the node returns a *CDER* packet that contains the length of data sent soon to the BS by setting the *next_hop* to the first node in the *neighbor_list*. Because the *neighbor_list* is in the ascending order of the receiving time of *CND* and *CNC*, the first node in the *neighbor_list*, sometimes even the second and the third and so on, must be one-hop close to the BS than the node itself. If the node is the neighbor of the BS, the BS must be the first node in its *neighbor_list*.

In packet *CDER*, we add the length of data field *data_length*, time field T_S, and random number field R_S, and the source address is set to itself, and the destination address is set to the BS. The packet *CDER* contains following information:

(1) the address of the source node, the previous hop, and the next hop,

(2) the length of data,

(3) the authentication information: time field T_S, random number field R_S.

Source node selects a broadcast key *BK* to encrypt the packet *DER*, selects the session key *SK* to encrypt $T_S \parallel R_S$, and broadcasts a confidential *DER* (*CDER*) packet to the whole network. After receiving this packet, each intermediate node does as follows (see Table 6):

(1) decrypt $DER = D_{BK}(CDER)$ with the broadcast key *BK* of the node;

(2) change the address *prev_hop* to the address of itself;

(3) get the address *prev_hop* in the first record of *neighbor_list* to set the *next_hop*;

(4) check if the broadcast packet has been received by searching *pkt_seq_num* in the table *rc_pkt_table*. If the packet has already been received once, the node drops this *CDER* and does not rebroadcast it. Otherwise, it stores *pkt_seq_num* in table *rc_pkt_table*, encrypts $CDER = E_{BK}(DER)$ with the broadcast key *BK*, encrypts $T_S \parallel R_S$ with the session key *SK,* and rebroadcasts the packet *CDER* to its neighbor.

When the BS receives the *CDER* packet, it decrypts $DER = D_{BK}(E_{BK}(CDER))$ with the broadcast key *BK* and $T_S \parallel R_S = D_{SK}(E_{SK}(T_S \parallel R_S))$ with the session key *SK*. It gets the *data_length* for the following calculations of the shortest path, through which we can get the energy for sending the data from the source node and random number field R_B and time T_B for authentication.

3.2.2. Routing Path Selection System (RPSS). After the BS receives the packet *CDER*, in order to tell the source node a best routing path to BS, it starts to calculate the shortest path to the node. However, it is important for BS how to calculate the shortest path in WSNs. We design routing path selection system (RPSS) to solve the problem.

As mentioned above, the shortest path has the minimal sum of energy consumed for transmitting one packet, that is, usually the path with minimum hops. Thus it saves the energy from the view of the whole network. When there are more than two shortest paths, we use the maximal available power as the second criteria; that is, we select the path that has the maximal available energy on each sensor node.

To get the desired shortest path, we modify the breadth first search (BFS) algorithm [5] to get the relatively shortest

Table 6: Confidential data enquiry reply packet forwards.

Source node (sender)		Intermediate nodes (receiver)
$DER = [pkt\_type \Vert source\_add \Vert data\_length$ $\Vert prev\_hop \Vert next\_hop \Vert pkt\_seq\_num]$; $CDER = E_{BK}(DER\ ) \Vert E_{SK}(T_S \Vert R_S)$	$\xrightarrow{\text{CDER}}$	$DER = D_{BK}\ (\ E_{BK}(CDER))$; $itself\_address \rightarrow\ prev\_hop$; *First record of neighbor_list* $\rightarrow\ next\_hop$; $pkt\_seq\_num\ \rightarrow\ rc\_pkt\_table$; $DER = [pkt\_type \Vert source\_add \Vert data\_length$ $\Vert prev\_hop \Vert next\_hop \Vert pkt\_seq\_num]$; $CDER = E_{BK}(DER\ )) \Vert E_{SK}(T_S \Vert R_S)$
		The base station
	$\xrightarrow{\text{CDER}}$	$DER = D_{BK}(E_{BK}(CDER\ ))$ $T_S \Vert\ R_S = D_{SK}(E_{SK}(T_S \Vert R_S))$; *Get data_length.*

path from the BS to source node, as is shown in Algorithm 1. The BFS always finds the shortest path from the source to the destination, if there is one. Our modified version of BFS algorithm does not necessarily select the absolute shortest path because we also need to consider the left energy, that is, the weight corresponding to each edge, of each node into consideration. That is, if one node on the shortest path has energy left less than required level, we discard this shortest one and continue searching the second shortest path until success.

We assume that the BS wants to get data from source node N. We first define three levels of energy limitation. Each level is the half of the upper level. The main modification to the breadth-first search algorithm is that whenever it finds a shortest path to source node N, it checks if the weight of each edge on the path is greater than the predefined level. If so it returns this path as the shortest path. Otherwise, it continues the calculations until it finds the second shortest path. If the shortest path under current energy limitation cannot be found, it means that each path found has at least one node whose energy level is less than current energy limitation. Consequently, we degrade the energy limitation to the lower level and search again. If not any path is found from the first level to the third level, it means that source node N is unreachable [20, 21].

In a word, BS maintains an energy limitation array for all nodes, and the updating of energy limitation for each node is independent. This feature ensures the best use of each node in the sensor network. In the RPSS, BS can determine whether it has the routing path to the source node or not and how many routes it may be selected. If there are routes to the source, the BS will select the shortest route and send to it in time.

3.2.3. Sending Routing Information. In the algorithm *multi_shortest_path*, we introduce into multiple-threaded process mechanism. As is to know that a thread is a lightweight process which exists within a program and executed to perform a special task in operating system. A process that has only one thread is referred to as a single-threaded process, while a process with multiple threads is referred to as a multiple-threaded process [22]. In our design, a thread is placeholder information associated with a single use of a program that can handle multiple concurrent users, and several threads of execution may be associated with a single process. In runtime environment designed by us, some threads exist in a common memory space and can share both data and code of a program, and they can increase the speed of any application.

We then present the process of executing the related function of RPSS. When any standalone application is running, it first executes the method *main* running in a one thread, called the main thread. The main thread creates another child thread which executes the function *send_route_to_source_node*, and the function schedules another function *multi_shortest_path*, which urgently creates another child thread which executes the function *send* when finding a route to the source node. Algorithm 2 shows code segments for sending route to source node algorithm. The method *main* execution can be finished, but the program will keep running until all threads have completed its execution. As is shown in Algorithms 1 and 2.

The multiple-threaded process mechanism mentioned above evidently decreases the delay with the multiple-threaded process because it can satisfy with multiple users and concurrent requests. If multiple users are using the function *multi_shortest_path*, the threads are created and maintained for each of them. Our design not only increases the speed of selecting a path to the source but also always saves memory space.

Once the BS has got the routing path to the source, the route is set to the field *route_list* of the packet RR. The BS then selects a broadcast key *BK* to encrypt the packet route reply (RB) including the route to the source, selects the session key *SK* to encrypt T_S and $R_S - 1$, and sends the confidential *RR* (*CRR*) packet to the second address of the route. Each intermediate node forwards this packet according to the corresponding of the route.

When the source node receives the *CRR*, at first, it must authenticate communication time cost by comparing the time field T_B with the current time and also authenticate the freshness of the packet by comparing random number field R_B with the foregone random number R_B. If the authentication fails, the source node can conclude that the sender of the packet is not real or the route is not credible and drops the packet. Otherwise, the source node stores *route_list* in table *route_table* and *pkt_seq_num* in table *rc_pkt_table*.

Multiple shortest path algorithm

```
vector<nsaddr_t> BEARP::multi_shortest_path(nsaddr_t
source_node, int packet_energy){
    bool visited[NODES_AMOUNT];
    nsaddr_t father[NODES_AMOUNT];
    nsaddr_t tmp, neighbor_node;
    vector<nsaddr_t> queue, route [PATHS_AMOUNT];
    bool found;
    int pointer, path_amount;

    found = false;
    path_amount = 1;
    nsaddr_t father_node;
    queue . reserve(NODES_AMOUNT);
    route . reserve(NODES_AMOUNT);

    while (!found){
        pointer = 0;
        for (int i = 0; i <NODES_AMOUNT; i++){
                visited[i] = false;
                father[i] = −1;
        }
        for (int i = 0; i <queue.size(); i++)
            queue · pop_out();

        visited[BASE_STATION] = true;
        queue . reserve(NODES_AMOUNT);
        queue . push_in((nsaddr_t)BASE_STATION);

        for (int i = 0; i <NODES_AMOUNT; i++){
            if (left_energy[i]− packet_energy >=
                current_energy_limit[source_node]){
                for (int j = 0; j<NODES_AMOUNT; j++){
                    if (weight_matrix[i][j] > 0 && weight
                          _matrix[i][j] < MAX_INT){
                      if (left_energy[j]− packet_energy >=
                           current_energy_limit[source_node])
                           all_neighbor_list[i] . push_in(j);
                      }
                }
            }
        }


        tmp = queue[pointer];
        visited[tmp] = true;
        while(queue · size() > pointer){
            for(int i = 0; i<all_neighbor_list[tmp] . size(); i++){
                neighbor_node = all_neighbor_list[tmp][i ];

                if (visited[neighbor_node])
                continue;
                father [neighbor_node] = tmp;

                if (neighbor_node == source_node){
                    father_node = neighbor_node;
                    while(father_node!= BASE_STATION){
                        route . push_in(father_node);
                        father_node = father[fa];
                    }
                    found = true;

                    if(path_amount = 1){
                      path_amount++;
                      threadbegin /* create a thread to send route*/
                      send(route[0], source_node);
```

Algorithm 1: Continued.

```
                      threadend
                    }
                }
                visited[neighbor_node] = true;
                queue . push_in(neighbor_node);
            }
            pointer++;
            tmp = queue[pointer];
        }
        if (found && path_amount = =PATH_MAX)
            break;
        update_energy_limit(source_node,
                    current_energy_limit[source_node]);
    }
    return route;
}
```

Algorithm 1: Code segments for multiple shortest path algorithm.

```
Send_route_to_source_node(nsaddr_t source_node, int packet_energy){
    /*start multi_shortest_path algorithm*/
    vector<nsaddr_t> BEARP::multi_shortest_path(nsaddr_t source_node, int packet_energy);
    bool exchange;
    int high, low, path_amount;
    multi_shortest_path (nsaddr_t source_node, int packet_energy);
    count(int timer); exchange = true;
    high = path_amount; low = 1;
    while(timer >= TIME_MAX && listening(ACK) = false){
        if(high >= low){
            if(exchange){
                exchange = false;
                path_amount −−;
                send(route[high −−], source_node);
            }
}
            else{
                exchange = true;
                path_amount −−;
                send(route[low + + ], source_node);
            }
        }
        else {
            multi_shortest_path(nsaddr_t source_node,
              int packet_energy);
            high = path_amount; low = 1; exchange = true;
        }
    }
    Return(1);
}
```

Algorithm 2: Code segments for sending route to source node.

Table 7 shows the process of forwarding the confidential route reply packet in the WSN.

At the same time, the *ACK* mechanism can also help the source node find a correct route to the BS. On receiving the *CRR* packet, the source node knows which path it can use to communicate with the BS. As a result, using the path transferred with the *CRR* packet, it returns an *ACK* packet to the BS to confirm the receipt of the *CRR*. The *ACK* packet also contains the number of data packets going to be sent, which to some extent guarantees that the receiver can detect the loss

Table 7: Confidential route reply packet forwards.

BS (sender)		Intermediate nodes (receiver)
$DER = D_{BK}(E_{BK}(CDER))$; $T_S \Vert R_S = D_{SK}(E_{SK}(T_S \Vert R_S))$; $RR = [pkt\_type \Vert source\_add \Vert route\_list \Vert pkt\_seq\_num]$; $CRR = E_{BK}(RR) \Vert E_{SK}(T_S \Vert R_S - 1)$.	$\xrightarrow{\text{CRR}}$	$RR = D_{BK}(E_{BK}(RR))$; *The corresponding of route_list* → *next_hop*; *pkt_seq_num* → *rc_pkt_table*; $CRR = E_{BK}(RR) \Vert E_{SK}(T_S \Vert R_S - 1)$.
		Source node
	$\xrightarrow{\text{CRR}}$	$RR = D_{BK}(E_{BK}(CRR))$; *route_list* → *route_table*; *pkt_seq_num* → *rc_pkt_table*; *authenticate* $T_S \Vert R_S - 1$: $T_S \Vert R_S - 1 = D_{SK}(E_{SK}(T_S \Vert R_S - 1))$.

of data due to communication problems, nodes failure, or misbehavior of compromised nodes. After sending the *ACK* packet, source node is ready to start transmitting the real data. If the BS does not receive the *ACK* packet within a predefined time, it deems the selected route as invalid and runs the function *multi_shortest_path* once more to find another path. If it receives the *ACK* packet, then it knows that this route is available and waits for data from the source node.

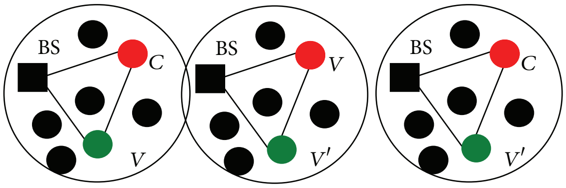


Figure 2: Network of including malicious nodes.

3.3. Routing Maintenance and IDS. In route maintenance, it is still the BS that works as the server to operate intrusion detection system (IDS) and to release control information. The purpose of the phase route maintenance is to overcome this potential risk by IDS and to prolong the network lifetime as much as possible [16–18, 23].

The BS must verify all nodes entering the network at the beginning. This is one of our assumptions. In our proposed solution, the BS detects any compromised node which possibly exists in the network by impersonating regular users. The BS detects the compromised node by sending each node arbitrary route requests one by one. Figure 2 shows a network of including compromised nodes. When a BS wants to test whether a node (let us say node C) is forwarding other nodes' data packets inside the network or not, the BS will first pick a validated destination node V that is close to node C. Then, the BS will send a *REQ* to node C for node V. Once node C agrees to participate in the route and a route is established between the BS, node C, and node V, the BS will send data packets to node V using this route. Then, by sending information and asking for the received packets to node V encrypted with its private shared key between the BS and node V, the BS will check whether node V has received the packets or not. Node V will send back an acknowledgment to BS whether it has received any data packets from node C or not. If it has not, the BS will test whether the node C and V are forwarding other nodes' data packets inside the network or not, so the BS will continue to pick another validated destination node V' and node V. Thus, the BS will check whether node V' has received the packets or not by sending information and asking for the received packets to the validated destination node V' encrypted with its private shared key between the BS and node V'. If it has not, the BS will mark node C or V as compromised nodes and will update the network key immediately. Algorithm 3 shows code segments for detecting compromised node algorithm in intrusion detection system.

All nodes in the network except for the compromised nodes will receive the new network shared key. From that the compromised nodes will not be able to encrypt or decrypt any packet information [24]. The BS will know that the nodes are compromised and take them out of the network. At the same time, the BS will process the routing tree including the compromised nodes.

4. Security Analysis of BEARP

In this section, we will analyze the security properties of BEARP required by sensor networks and present how BEARP defends some typical attacks in the WSN.

4.1. Routing Message Confidentiality. A sensor network should not leak sensor readings, especially control packet, to neighboring networks. We have assumed that the key management system is secure, which is the underlying security for our BEARP, so the secret keys are confidential. The standard approach for keeping sensitive routing message secret is to encrypt them with a secret key that only intended receivers possess, hence our BEARP can distinctly achieve routing message confidentiality. Given the observed communication patterns, we set up secure channels between nodes and base stations and later bootstrap other secure channels as necessary.

4.2. Identity Authentication and Routing Message Authentication. Since an adversary may exert to personate or imitate a compromised node, identity authentication and routing message authentication are important for many applications in sensor networks [25]. The receiver needs to ensure that

```
Detect_compromised_node(compromised node $C$, validate node $V$ )
/* Use the validate node $V$ to confirm whether the node $C$ is compromised node or not.*/
        { if (Base Station received the acknowledge packet of the validate node)
              return (true);
          else {
      select another validate node $V'$;
          if (detect_compromised_node(compromised
              node $V$, validate node $V'$))
            validate node $C$ is compromised node;
            else
            {
          detect_compromised_node(compromised
            node $C$, validate node $V'$);
          validate node $V$ is compromised node;
          }
      }
  }
```

Algorithm 3: Code segments for detecting compromised node algorithm.

the routing message used in any decision-making process originates from a trusted source. Informally, routing message authentication allows a receiver to verify that the routing message was really sent by the claimed sender. In the two-party communication case of the BEARP, routing message authentication can be achieved through a purely symmetric mechanism: the sender and the receiver share a session secret key to compute the four particular parameters (T_B, R_B, T_S, and R_S) of all communicated routing message because they are correlative with the routing message. When a routing message with four correct particular parameters arrives, the receiver knows that it must have been sent by the sender. If the four particular parameters have some mistakes, the receiver concludes the sender or intermediate node may be adversary.

4.3. Routing Message Integrity. In communication, routing message integrity ensures the receiver that the received routing message is not altered in transmission by an adversary [5]. In BEARP, we achieve routing message integrity through routing message authentication for the four particular parameters (T_B, R_B, T_S, and R_S), which is not a stronger property. It is very difficult that an adversary only alters routing information but does not alter the four particular parameters because the routing message is confidential as a whole. At the same time, the packet sequence number *pkt_seq_num* can also help authenticate routing message integrity.

4.4. Routing Message Freshness. Routing message freshness means that the routing message is recent, and it ensures that no adversary replayed old messages. Sensor networks send measurements over time, so it is not enough to guarantee confidentiality and authentication [5]. In BEARP, to ensure each routing message is fresh, we design a real-time T_B or T_S field of the routing packet, which provides to conclude the freshness of the packet through computing and comparing the two particular parameters (T_B, T_S), the receiving time, allowing delay time.

4.5. Defending Some Typical Attacks. The most direct attack against a routing protocol is to target the routing information exchanged between nodes. By spoofing, altering, or replaying routing information, adversaries may be able to create routing loops, attract or repel network traffic, extend or shorten source routes, generate false error messages, partition the network, increase end-to-end latency, and so forth. [9]. Apparently, routing information in BEARP which holds the above four security properties can defend adversaries to spoof, alter, or replay them.

Wormholes are hard to detect because they use a private, out-of-band channel invisible to the underlying sensor network. Sinkholes are difficult to defend against in protocols that use advertised information such as remaining energy or an estimate of end-to-end reliability to construct a routing topology because this information is hard to verify [26].

However, resistant to the two attacks is the most important of all secure targets of our designing the routing protocols [25]. Adversary cannot encrypt and decrypt the routing information with the secret key, and it cannot pretend to be another node to impersonate and fabricate any other information. Furthermore, all routing paths are selected uniquely by the BS which is very secure and cannot be compromised under any condition in our assumption. Therefore, protocols that construct a topology initiated by a BS are most susceptible to wormhole and sinkhole attacks, but BEARP can easily defend them.

In a selective forwarding attack, compromised nodes may intend to include themselves on the actual path of the data flow and refuse to forward certain messages and simply drop them, ensure that they are not propagated any further. However, once again the mechanism that BEARP selects routing paths prevents sensor nodes from selecting or joining routing path. All routing paths are selected uniquely by the BS, which defends adversaries to join in the WSN. A more subtle form of this attack is when an adversary selectively forwards packets. An adversary interested in suppressing or modifying packets originating from a select few nodes can reliably forward the remaining traffic and limit suspicion of its wrongdoing. Routing message confidentiality can prevent adversary to open any routing packets.

When an adversary captures a sensor node in WSNs and knows all its secret keys, BEARP also has two methods for secure process: one is routing paths selected uniquely by the secure BS, which reject a sensor node captured to imitate BS; another is IDS of detecting compromised nodes, which can take the sensor node captured out of the WSN.

5. Performance Evaluations and Analyses

The goal of our experiments is to evaluate and analyze the performance of our BEARP. To simplify the simulation, we generated random nodes and defined some of them as compromised nodes. In BEARP, these compromised nodes are kicked out of the network as soon as they discovered; however, in DD protocol and S-DD protocol, these nodes are not detected. In our simulations, no compromised nodes will be allowed to reenter the network before being certified by the BS, and therefore they will not be able to route packets again. In the following sections, We measured the packet delivery ratios, network lifetime, and nodes blocked by compromised attacks during data forwarding, which are very important for efficient and secure routing protocol for WSNs [3], and we then show the simulation results for different scenarios.

5.1. Simulation Metrics. To evaluate the performance of our secure routing mechanism in the presence of some compromised nodes which impact network performance, we have simulated BEARP on a network simulator, ns-2 [27]. In our simulations, we consider to generate a variety of sensor fields of different sizes. Some sensor nodes, ranging from 100 to 1200, are randomly deployed in $200 \times 200\,\text{m}^2$ target area, and the network size is changeable according to different measure for network performance. Regarding the left-bottom corner of the target area as (0, 0), we positioned a BS at a fixed point (100, 100), almost in the center of the WSN. Each sensor node has a constant transmission range of 20 m. All sensor nodes are stable, and no node is moving, and every round each node sends 20 packets to the BS. We changed the scenario files each time for testing the BEARP protocol, DD, and S-DD protocol for different numbers of nodes, compromised nodes.

5.2. Packet Delivery Ratios. In this scenario, we increased the compromised nodes into the WSN for every test case. The simulation time was 90 s in test. In Figure 3, we show the packet delivery ratio when there are some compromised nodes amounts from 10 to 100 present in the WSN. As we can see from the figure, the BEARP has better packet delivery ratio than the DD protocol and S-DD protocol all the time. This is due to the fact that since compromised nodes are left out of the network because of encryption and authentication in BEARP, all data packets may not be sent to them. Therefore, the packet delivery ratios of the BEARP hold rather higher than those of the DD protocol and S-DD protocol.

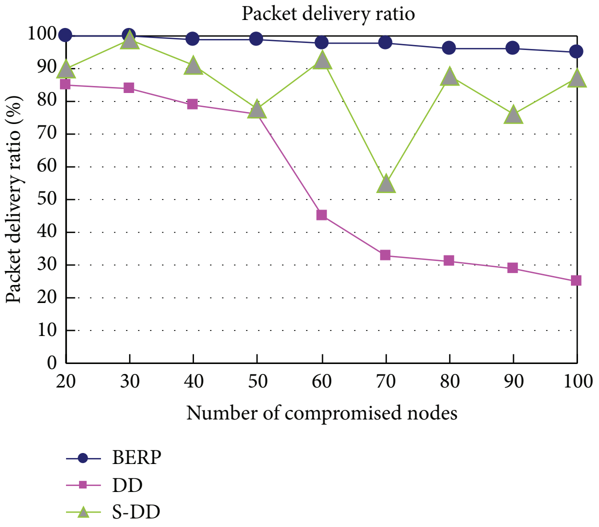


Figure 3: Packet delivery ratio (%) for 600 nodes.

At the same time, as the number of compromised nodes increases, the packet delivery ratio for both protocols goes down because the compromised nodes are dropping the packets. Especially, the packet delivery ratio for DD protocol descends sharply when the compromised nodes are more than 50. In S-DD protocol, without authentication mechanism between neighbors, the compromised node may transmit or not when the packets send to a compromised node. Thus, the packet delivery ratio for S-DD protocol is unstable. We think that since certain compromised nodes are chosen randomly, there is a chance that compromised nodes may occupy crucial positions for data transferring. In BEARP, there are not many nodes left in the WSN since the compromised nodes are being left out due to the mechanism of encryption and authentication. Therefore, it is taking a certain time to establish connections and for the packets to be delivered, and the packet delivery ratio for BEARP decreases slightly.

5.3. Network Lifetime. The most significant performance increase achieved in BEARP is the network lifetime. In Figure 4(a), we can see that BEARP increases the network lifetime over 15% and 8%, respectively, compared to DD protocol and S-DD protocol. Though the rule for reinforcing a particular path differs, it is always the fact that DD protocol and S-DD protocol use the same path for all communications between the same source and BS. The direct consequence is that nodes on this particular path may deplete energy very soon, while BEARP uses several shortest paths and maintains an energy limitation array for all nodes to avoid each node to exhaust energy quickly. Figure 4(b) is the simulation results for network lifetime when 10% of nodes misbehave. From this figure we can see that network lifetime of DD protocol suffers a significant decrease, and S-DD protocol's lifetime is increased but unstable while that of BEARP decreases slightly and be stable. When compromised nodes destroy the path for forwarding, both DD protocols and S-DD protocol

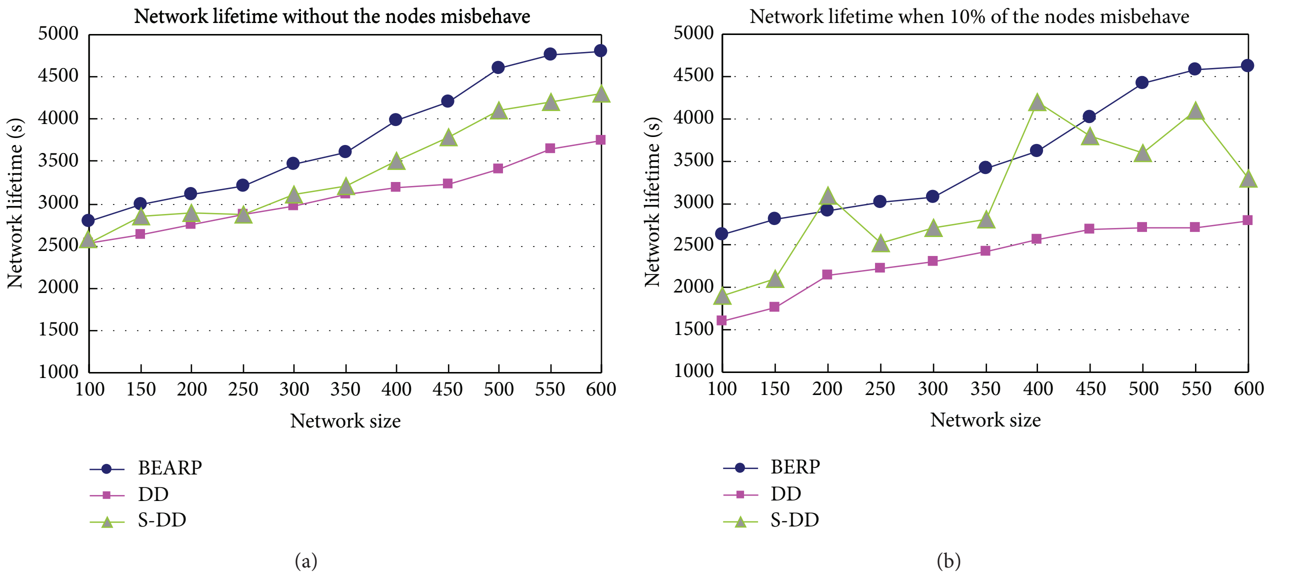


FIGURE 4: Compared average network lifetime between BEARP, DD, and S-DD protocol. (a) without the nodes misbehaving. (b) when 10% of the nodes misbehave.

have to select a new path, and the communication load spreads among a small number of available paths. Instability of lifetime for S-DD protocol is that it cannot detect and reject compromised nodes. Moreover, the lifetime of BEARP is 37% longer than DD protocol because BEARP can reject compromised node, resist their attacks, and distribute the load more evenly among several secure paths according to the algorithm *multi_shortest_path*.

5.4. Nodes Blocked by Compromised Attacks. We randomly distributed compromised nodes over the square area. In the simulations, we considered two types of compromised nodes: one drops all the relaying packets (type I), and the other drops all the relaying packets and also advertises inconsistent routing information (type II). In addition, we simulated two different density networks: one is 600 sensor nodes network, and the other is 1200 sensor nodes network.

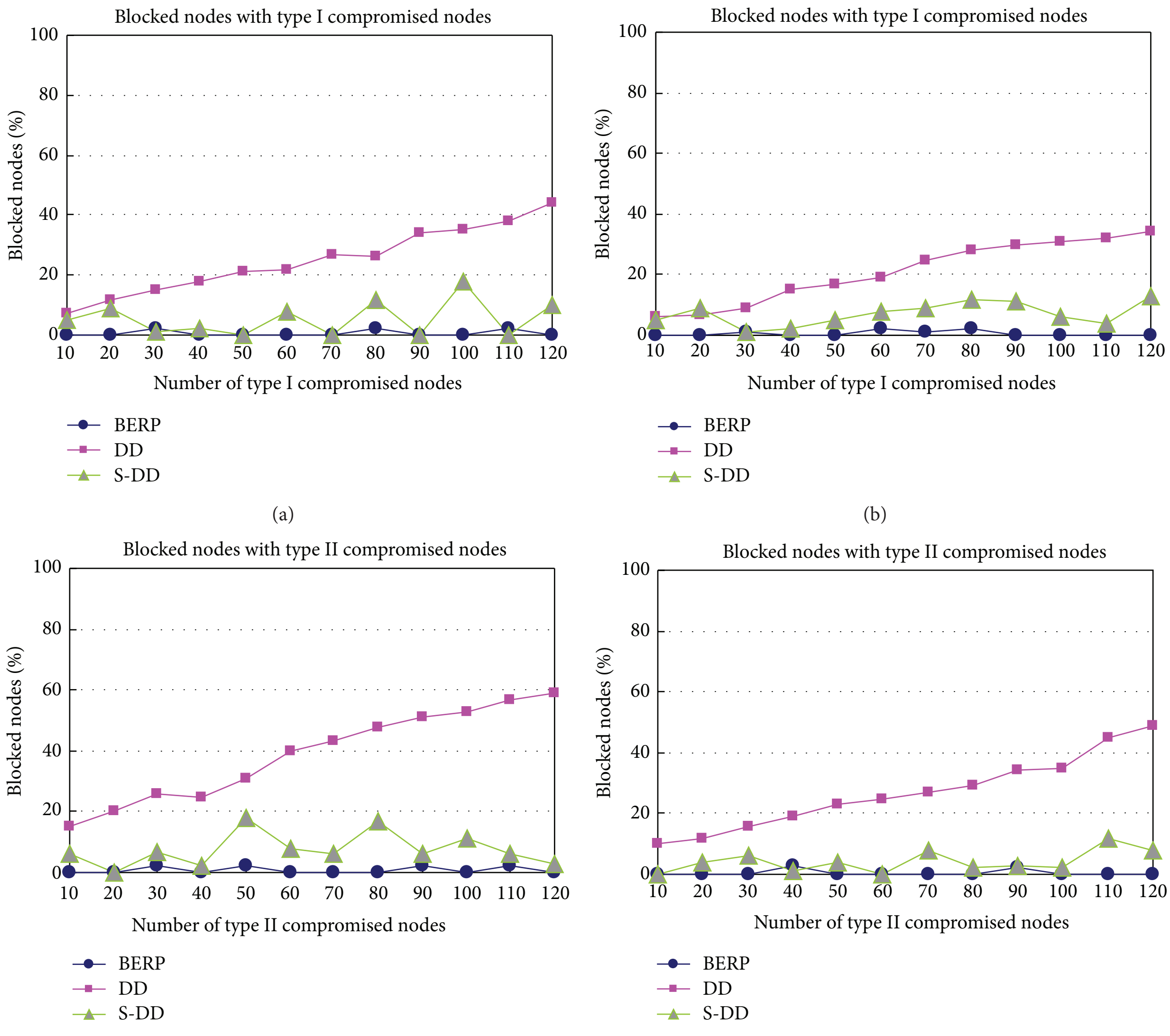
We performed a set of experiments to measure the number of sensor nodes blocked by a set of compromised nodes in each round, increasing the number of compromised nodes in the network. In the presence of type I compromised nodes, we, respectively, measured the number of blocked nodes running on the BEARP, on the DD protocol, and on the S-DD protocol in both 600 and 1200 sensor nodes networks. We also measured the number of blocked nodes using the same scheme in the presence of type II compromised nodes.

Each simulation experiment was conducted using 10 different network topologies, and each result was averaged over 10 runs of different network topologies.

Simulation experiment results are shown in Figure 5. In the presence of type I compromised nodes which drop all the relaying packets, the effect of DD protocol, S-DD protocol, and our BEARP on a ratio of blocked nodes is shown in Figures 5(a) and 5(b). S-DD protocol is not stable to be blocked by type I compromised nodes. In 600 sensor nodes network, using DD protocol incurs blocked nodes from about 5% to 44%, while BEARP has almost no blocked nodes for compromised node to be entered to WSNs due to the secure authentication mechanism, as shown in Figure 5(a). In Figure 5(b), in 1200 sensor nodes network, using DD protocol has less blocked nodes than in 600 sensor nodes network. This is because, the number of sensor nodes scattered in the network is doubled, which makes the network denser. Also, each sensor node has more neighbor nodes so that it has more next-hop nodes. Thus, this increases the chances of bypassing the compromised nodes which drop relaying packets.

In the presence of type II compromised nodes which both drop all the relaying packets and advertise inconsistent routing information, the effect of secure authentic system on a ratio of blocked nodes is shown in Figures 5(c) and 5(d). Without secure authentic system, the influence of type II compromised nodes over the network is more devastating than that of type I nodes, since type II compromised nodes even attract the network traffic and drop them. Using secure authentic system, however, we see that more than 99% of sensor nodes are not blocked, as shown in Figures 5(c) and 5(d). Since, in the experiments, almost every type II nodes were excluded by secure encryption and authentication system from the network; legitimate nodes did not forward packets to the compromised nodes identified. Thus, with several type II nodes, almost all of them are excluded from the network so that more than 99% of sensor nodes are not blocked.

Out of control is the cause of network blocked. In WSN with compromised nodes, DD protocol cannot control the relaying for any packets, while S-DD protocol cannot control the relaying of neighbor's packets due to no secure authentic system. On the one hand, our secure authentic system in BEARP can protect the sensor nodes from compromising



(a)

(b)

(c)

(d)

Figure 5: Performance evaluation for nodes blocked by compromised attacks (average over 10 runs). (a), respectively, executes BEARP, DD, and S-DD protocol in 600 sensor nodes network, in the presence of type I compromised nodes; (b), respectively, executes BEARP, DD, and S-DD protocol in 1200 sensor nodes network, in the presence of type I compromised nodes; (c), respectively, executes BEARP, DD, and S-DD protocol in 600 sensor nodes network, in the presence of type II compromised nodes; (d), respectively, executes BEARP, DD, and S-DD protocol in 1200 sensor nodes network, in the presence of type II compromised nodes.

in WSNs. On the other hand, even if the sensor nodes suffer insurmountable attacks of compromised nodes, the IDS including the algorithm *detect_compromised_node* has an ability to detect type I and type II compromised node, and makes them become no more a member of the network so that they cannot influence other legitimate nodes any more. However, seen in Figures 5(b) and 5(d), as the network gets denser and each node's degree becomes higher, our BEARP makes the network more resilient in the presence of type I or type II compromised nodes.

6. Conclusions

Nowadays, most of the wireless sensor network routing protocols are implemented with no security in mind. Incorporating security into these protocols can only solve some simple security problems, so we focus on security mechanisms for the WSN and design a security routing protocol as a goal, which is performed throughout the network. Simultaneity, using the power of network nodes for security is a necessary evil. Consequently, we propose the efficient and secure routing protocol called BEARP.

We presented the BEARP absolutely different from the well-known DD protocol and the other routing protocol incorporated security. BEARP can successfully not only achieve routing message confidentiality, authentication, integrity, and freshness but detect the compromised nodes in a network by IDS. We implemented an encryption and authentication mechanism to encrypt the data packets between any two nodes and authenticate BS and source node in the network. Moreover, BEARP has two methods for secure process: one is routing paths selected uniquely by the secure BS; another is our important IDS for detecting compromised node. All the secure mechanisms are united together to make our routing protocol BEARP effectively resilient in the presence of compromised nodes that launch selective forwarding attacks, wormhole attacks, sinkhole attacks, and even a node captured.

At the same time, we also make full use of the severely limited resource presented by WSNs, especially the energy limitation. Our BEARP mitigates the loads of sensor nodes by transferring routing-related tasks such as RPSS and IDS to the BS, which not only efficiently maintains network wide energy equivalence and prolongs network lifetime but also successfully improves our security mechanism. Especially, in algorithm *multi_shortest_path* of the RPSS, we design the multiple-threaded process mechanism, which not only increases the speed of selecting a path to the source but also always saves memory space and the contents of the register when RPSS is interrupted and restored. Furthermore, RPSS maintains an energy limitation array for all nodes, and the updating of energy limitation for each node is independent. This feature ensures the best use of each node's energy in the sensor network.

Simulation results show a favorable increase in the performance evaluation for BEARP when compared to DD protocol in the presence of compromised nodes. Our protocol surpasses the DD protocol and S-DD protocol in terms of the packet delivery ratios, network lifetime, and nodes blocked by compromised attacks during data forwarding.

However, we only considered the efficient and secure routing protocols of BS actively launch. In future work, we will focus the research on security mechanisms for other different WSNs, also for particular misbehaviors of some compromised nodes such as denial-of-service attacks and jamming attacks.

Acknowledgments

The authors would like to thank anonymous referees for their useful comments. This work is supported by the Special Research Funds of Educational Ministry of China under Grant no. 104086.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] 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.

[3] A. Perrig and R. Szewczyk, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 9, pp. 534–548, 2002.

[4] A. Quintero, D. Y. Li, and H. Castro, "A location routing protocol based on smart antennas for ad hoc networks," *Journal of Network and Computer Applications*, vol. 30, no. 2, pp. 614–636, 2007.

[5] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[6] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, "An on-demand secure routing protocol resilient to byzantine failures," in *Proceedings of the ACM Workshop on Wireless Security*, pp. 21–30, September 2002.

[7] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 56–67, Boston, Mass, USA, August 2000.

[8] X. Wang, L. Yang, and K. Chen, "SDD: secure directed diffusion protocol for sensor networks," in *Proceedings of the 1st European Workshop (ESAS '04)*, C. Castelluccia et al., Ed., Lecture Notes in Computer Science, pp. 205–214, August 2004.

[9] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[10] R. Roman, C. Alcaraz, J. Lopez, and N. Sklavos, "Key management systems for sensor networks in the context of the internet of things," *Computers & Electrical Engineering*, vol. 37, no. 2, pp. 147–159, 2011.

[11] L. Zhou and Z. J. Haas, "Securing wireless sensor networks," *IEEE Network Magazine*, vol. 6, pp. 30–37, 1999.

[12] P. Papadimitratos and Z. J. Haas, "Secure routing for mobile wireless sensor networks," in *Proceedings of the SCS Communication Networks and Distributed Systems (CNDS '02)*, pp. 27–31, 2002.

[13] N. Nasser and Y. Chen, "SEEM: secure and energy-efficient multipath routing protocol for wireless sensor networks," *Computer Communications*, vol. 30, no. 11-12, pp. 2401–2412, 2007.

[14] S. B. Lee and Y. H. Choi, "A secure alternate path routing in sensor networks," *Computer Communications*, vol. 30, no. 1, pp. 153–165, 2006.

[15] B. Wu, J. Wu, E. B. Fernandez, M. Ilyas, and S. Magliveras, "Secure and efficient key management in mobile ad hoc networks," *Journal of Network and Computer Applications*, vol. 30, no. 3, pp. 937–954, 2007.

[16] Y. C. Hu, A. Perrig, and D. B. Johnson, "Rushing attacks and defense in wireless Ad Hoc network routing protocols," in *Proceedings of the ACM Workshop on Wireless Security*, pp. 30–40, New York, NY, USA, September 2003.

[17] Y. C. Hu, D. B. Johnson, and A. Perrig, "SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 175–192, 2003.

[18] J. Deng, R. Han, and S. Mishra, "INSENS: intrusion-tolerant routing for wireless sensor networks," *Computer Communications*, vol. 29, no. 2, pp. 216–230, 2006.

[19] T. Ito, H. Ohta, N. Matsuda, and T. Yoneda, "A key predistribution scheme for secure sensor networks using probability density function of node deployment," in *Proceedings of*

the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '05), pp. 69–75, November 2005.

[20] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 255–265, August 2000.

[21] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensornetworks," *Communications of the ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[22] G. J. Narlikar, "Scheduling threads for low space requirement and good locality," *Theory of Computing Systems*, vol. 35, no. 2, pp. 151–187, 2002.

[23] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, pp. 1987–1997, April 2003.

[24] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 52–61, October 2003.

[25] H. Yang, F. Ye, Y. Yuan, S. Lu, and W. Arbough, "Toward resilient security in wireless sensor networks," in *Proceedings of the 6th ACM International Symposium on Mobile (MobiHoc '05)*, pp. 34–45, 2005.

[26] M. Abomhara, O. Zakaria, O. Khalifa, A. Zaidan, and B. Zaidan, "Enhancing selective encryption for H. 264/AVC using advanced encryption standard," *International Journal of Computer Theory and Engineering*, vol. 2, no. 2, pp. 223–229, 2010.

[27] G. Chen, J. W. Branch, M. Pflug, L. Zhu, and B. K. Szymanski, "SENSE: a wireless sensor network simulator," in *Advances in Pervasive Computing and Networking*, B. Szymanski and B. Yener, Eds., Springer, New York, NY, USA, 2005.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 716265, 14 pages
http://dx.doi.org/10.1155/2013/716265

Research Article

A Hybrid Authenticated Group Key Agreement Protocol in Wireless Sensor Networks

Yue Li,[1] Dehua Chen,[1] Wei Li,[1] Gaoli Wang,[1] and Paul Smith[2]

[1] *School of Computer Science and Technology, Donghua University, No. 2999 North Renmin Road, Songjiang, Shanghai 201620, China*
[2] *Computer Games Academic Team EB.3.13A, Docklands Campus, ADI University of East London, London E16 2RD, UK*

Correspondence should be addressed to Yue Li; frankyueli@dhu.edu.cn

Received 14 October 2012; Revised 28 February 2013; Accepted 1 March 2013

Academic Editor: Motoki Sakai

Wireless sensor networks are a modern and advanced technology whose applications are fast developing in recent years. Despite being a fascinating topic with various visions of a more intelligent world, there still exist security issues to be resolved in order to make WSNs fully adoptable. Due to the resource constraints of sensor nodes, it is infeasible to use traditional key establishment techniques that find use in fixed communication systems. In this paper, the design of a new hybrid Authenticated Group Key Agreement (AGKA) protocol is described for WSNs. The AGKA protocol reduces the high cost public-key operations at the sensor side and replaces them with efficient symmetric-key based operations. The proposed AGKA protocol is not only efficient but also meets strong security requirements. In order to demonstrate the protocol is verifiably secure and trustworthy, a formal verification of the AGKA protocol is carried out. Furthermore, several experiments are conducted on MICAz and TelosB platforms in order to evaluate the performance of the proposed protocol. The evaluation results show that the AGKA protocol is well suited for use with resource-constrained sensor nodes.

1. Introduction

Wireless sensor networks (WSNs) are viewed as a large number of small sensing self-powered devices/nodes which gather information or detect special events and communicate in a wireless fashion, with the end goal of handing their processed data to a base station. A diverse set of applications for sensor networks encompassing different fields have already emerged including medicine, agriculture, environment, military, electrical power systems, home appliances, toys, and many others.

In these and other vital, life-critical, or security-sensitive applications, secure and fast transmission of sensitive digital information over the sensor network is essential. A solid key management framework is one of the most crucial technologies for achieving secure infrastructure in wireless sensor networks.

Considering the limited resources of both computational ability and power supply of wireless sensor devices, the design of security protocols for wireless sensor networks is a nontrivial challenge given that most public key operations require expensive computations. Therefore, there is a need to employ energy-efficient key agreement protocols in order to prolong each sensor's battery life.

In recent years, symmetric-key-based key establishment schemes have gained popularity due to their small computational overhead. A promising solution for the establishment of symmetric keys in wireless sensor network applications is to use key predistribution protocols such as those studied in various papers [1–3]. Although symmetric mechanisms achieve low computational overhead when compared with public key operations, the key management for symmetric key based protocols is complicated and is always subject to attack by adversaries. Therefore, many public-key-based protocols have been proposed [4–11] for wireless sensor networks which give more flexibility and scalability.

In this paper, we focus on WSN applications involving clusters of wireless sensor nodes. We have designed a new hybrid authenticated group key agreement (AGKA) protocol. The motivation of which was to exploit the difference in capabilities between gateways and sensors and put the cryptographic burden on gateways where the resources are less constrained. We have also implemented the AGKA protocol on

TelosB and MICAz motes and performed several experiments in order to evaluate the performance of the AGKA protocol in terms of its energy consumption and memory usage. The evaluation results show that the proposed protocol is well suited for use with resource-constrained sensor nodes with limited processing power and power resources.

The remainder of this paper is organized as follows. Section 2 describes related works. Some preliminaries and network model are reviewed in Section 3. Section 4 presents our key agreement protocol. In Section 5, the security of the proposed protocol is discussed. We present the performance evaluations in Section 6 and provide our research conclusions in Section 7.

2. Related Works

SPINS [12] is one of the most popular symmetric-key-based security schemes used today. In this memory-efficient scheme, the nodes need only share a key with the base station, and establish keys with other nodes through the base station. This type of scheme is suitable for sensor networks with small numbers of sensor nodes manually deployed around the base station. The big drawback of this scheme is that the base station is a single point of attack, which could result in the compromise of the entire network. Those nodes closest to the base station must forward a high volume of traffic to the base station and this reduces the lifetime of the network as these nodes expend greater energy resources.

Key predistribution is an alternative approach, which distributes the keys to all sensors prior to the deployment of the sensors. Zhu et al. [13] proposed Localized Encryption and Authentication Protocol (LEAP) which supports the establishment of four types of keys for each sensor node including a pair-wise key and a group key (a network-wide shared key).

Eschenauer and Gligor [1] proposed the use of random graph theory, which was used to develop one of the first random predistribution schemes. A random graph is fully connected with a high probability if the average degree of its nodes is above a certain threshold. Generally high-density deployments result in a fully connected network. Hence, key establishment only needs to be performed such that any two neighbors have some probability p of successfully completing key establishment. Eschenauer and Gligor used this theory to develop a framework for key random predistribution protocols. This framework involves three phases: predistribution, shared-key discovery, and path-key establishment.

The computation complexity and energy consumption of those symmetric-key-based protocols are relatively small. However, the key management for pure symmetric-key-based systems can be complicated, a key distribution center (KDC) can be required, or a large number of symmetric keys can be preloaded into devices. Both of these solutions can reduce the scalability of WSNs. In contrast, public-key-based protocols give more flexibility and scalability in large sensor networks where new devices keep entering the cluster. However, public-key-based protocols require more expensive computational power.

In cluster-based wireless sensor networks, the design of secure group key establishment protocols is a foremost security issue. A group key establishment protocol allows participants to construct a group key that is used to encrypt/decrypt transmitted messages among participants over an open channel.

Recently several key agreement protocols have been proposed to offload public-key cryptographic computational requirements to servers and have the low-end devices do less work. Bresson et al. [4] proposed a group key agreement protocol well suited to imbalanced wireless networks consisting of devices with strict energy consumption restrictions and wireless gateways with less stringent restrictions. Their idea was to let a cluster of mobile devices and one wireless gateway dynamically agree on a session key. However, their protocol does not satisfy some important security properties such as mutual authentication and forward secrecy [14].

Nam et al. [15] further improved the mutual authentication of Bresson et al.'s protocol by adopting the Katz-Yung scalable compiler [16] whereby one online signature and $n-1$ verifications must be required; the computational cost, though reduced, is still expensive for resource constrained sensor devices.

Tseng [17] proposed an efficient group key agreement protocol based on the two aforementioned protocols. It employs an online/offline signature scheme [18] and shifts much of the computation to the wireless gateways possessing more computational power and energy. Nevertheless, it does not satisfy some important security properties such as mutual authentication [19].

In recent years, Elliptic-Curve-Cryptography-based-key agreement protocols [5, 9, 10, 20–22] have been designed for use in constrained mobile device environments and wireless sensor networks because of their small key sizes, such as the ECMQV protocol with ECC X.509 certificates [20] and implicit certificates [21] and the ECDSA authenticated key exchange protocol [22]. In 2004, Huang et al. proposed a hybrid authenticated key establishment protocol based on probably secure elliptic curve encryption [5] and the elliptic curve implicit certificate scheme [20]. In 2005, Liu and Ning created TinyECC [23], a software package that provides Elliptic Curve Cryptography (ECC) operations for TinyOS [24]. It supports all elliptic curve operations over prime fields F_p, including point addition, point doubling, and scalar point multiplication, as well as ECDSA operations. In 2011, Ammayappan et al. proposed an ECC-based two-party authenticated key agreement protocol for mobile ad hoc networks, which utilises both RSA and ECC to achieve mutual authentication. This method increases the computation burden on sensor side [10].

Using the concept of Schnorr Signature [25] and based on ECC, Huang et al. in [5] designed a key establishment in the authentication procedure of the access control scheme for WSNs. The new designed key establishment in [11] also used the concept of "timebound" in which once time period has elapsed, the sensor node in the wireless sensor network cannot access any data for a future time period in order to protect future messages. Huang et al. claimed that the authentication procedure and common key generation proposed in [5] offers computational efficiency, energy, and bandwidth savings. Nevertheless, adversaries can still apply a sensor node replication attack in the period of the expiration time. The reason is that the adversary can compromise the sensor node and apply the replication attack before expiration time.

In order to reduce communication cost, some ID-based protocols for wireless sensor networks have been proposed where a sensor node does not need to transmit its implicit certificate [8]. Zhang et al. proposed three protocols for wireless sensor networks [6, 26, 27]. Those protocols offer low communication overhead and low memory requirements by eliminating the public key certificate. But in those protocols, sensor nodes should still perform expensive computation such as Weil/Tate pairing and Map-to-Point operations. Recently, Zhang et al. [8] proposed an efficient ID-based protocol for key agreement in wireless sensor networks. This protocol removes expensive operations from a sensor node side and eliminates the communication overhead of transmitting public-keys, but this protocol is vulnerable to replication attacks, where adversaries can use this weakness to masquerade as a security manager and share the pair-wise key with the sensor node.

From the discussion of the recent representative key agreement protocols designed for wireless sensor networks, we find that those protocols are computationally expensive for sensor nodes or vulnerable to impersonator's attacks. It can be seen that the design of a secure authenticated group agreement protocol well suited to wireless sensor networks is a nontrivial challenge, which inspires us to propose a verifiably secure authenticated group key agreement protocol.

3. Network Model and Notations

Before the discussion of key establishment protocols involving public key cryptography, we will first present the model of the unbalanced cluster-based wireless sensor networks.

3.1. Network Model. The IEEE 802.15.4 low-rate wireless personal area network standard [28] specifies the physical layer and medium access control layer of a low data rate, ultra low power, and low cost sensor network. It defines two device types: a Full Functional Device (FFD) and a Reduced Functional Device (RFD). An RFD takes on the role of an end device, such as a low-power sensor, while an FFD takes the role of a coordinator, a gateway, or a security manager.

The wireless system environment we model is an unbalanced/asymmetric cluster-based wireless sensor network, which consists of some sensor nodes with strict computational capability restrictions and a gateway with less restriction. We consider a set of resource-limited sensor nodes (also called low-power nodes) communicating with a gateway (also called powerful node), in which each low-power node can send messages to the gateway via unicast communication, and the gateway can broadcast or unicast messages to each low-power node. The gateway covers an entire group region called a cell. It is the cluster-head of the group region. In the group region, the data transmission between gateway and its client nodes uses low-power wireless technology such as IEEE 802.15.4 standard and Zigbee. The communication between gateways and the base station could use WiFi and wired LAN technology. The monitoring software on the base station can collect and analyze the sensing data and put the useful information on the web server. All the authenticated users can login to the website to not only get the information of the target object but also maintain the sensor network by performing tasks such as updating/renewing the group key, putting a particular group of sensor nodes into sleep mode or merging the neighboring groups.

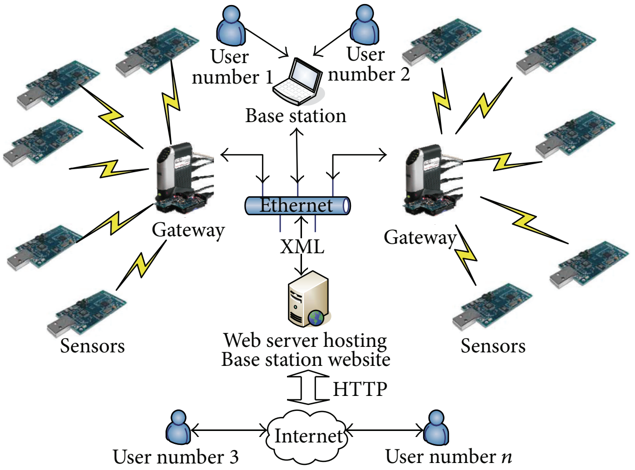


Figure 1: Network model of the asymmetric wireless sensor network.

Figure 1 shows the network model of the asymmetric wireless sensor network.

3.2. Key Notation and Terms. Let $U = 1$ be the initial set of low-power sensor nodes that want to generate a group key with gateway V. In Table 1, we summarize the key notations and terms used in the group key agreement protocol.

4. The Proposed Group Key Agreement Protocol

This section specifies the algorithms and features of the proposed AGKA protocol. The new AGKA protocol is implemented using the elliptic curve version of the Diffie-Hellman problem [29]. In addition to the use of an ECC cryptosystem, the proposed AGKA protocol also adopts a symmetric-key cryptosystem. The protocol reduces the cost of elliptic curve random point scalar multiplications at the sensor side and replaces them with low cost and efficient symmetric-key-based operations. Furthermore, it authenticates the entities based on a combination of the Elliptic Curve Digital Signature Algorithm (ECDSA) [30] and the Message Authentication Code (MAC).

The AGKA protocol consists of four algorithms.

(1) The key generation algorithm $AGKA.Kgen(\ell)$ is a probabilistic algorithm which on input of a security parameter ℓ provides each client $U_i \in \psi_C$ and the gateway with long-lived keys.

(2) The setup algorithm $AGKA.Setup(\vartheta)$ is an interactive protocol which on input of a set of clients ψ_C sets the

Table 1: Key notation and terms.

Notation	Description
ψ_C	Set of clients
q	A large prime
p	A large prime such that $p = 2q + 1$
P	Denotes a base point of large order n selected for an elliptic curve, which is public to all users
g	A generator for the subgroup G_q
(Q_i, q_i)	Public key and private key pair of a low-power node U_i, $Q_i = g^{q_i} \bmod p$
(Q_V, q_V)	Public key and private key pair of the powerful node V
(D_i, d_i)	Ephemeral public key and private key pair of low-power node
$\mathrm{Sig}_{U_i}(m)$	The signing algorithm based on ECDSA schemes under U_i's private key q_i and the signed message m
$\Vert$	Denotes concatenation
N	Nonce
c	Counter
MAC(M, K)	The computation of a MAC for a message M using MAC key K
GK	Group shared session key

wireless client group to be $\psi_C = \vartheta$ and provides each client U in ψ_C with a secret value sk shared with the gateway.

(3) The join algorithm *AGKA.join*(ϑ) is an interactive protocol which on input of a set of clients ϑ updates the wireless client group ψ_C to be $\psi_C \cup \vartheta$ and provides each client U in ψ_C with a (new) shared secret value sk.

(4) The remove algorithm *AGKA.Remove*(ϑ) is an interactive protocol which on input of subset ϑ of the wireless client group ψ_C updates the latter to be $\psi_C \setminus \vartheta$ and provides each client U in ψ_C with a new shared secret value sk.

Each cluster/group in a hierarchical cluster-based WSN is represented as the set μ, which consists of N sensor devices (also called clients), and a gateway. A nonempty subset of μ is called sensor client group ψ_C, which consists of clients communicating with the gateway. An elliptic curve E defined over prime fields $\mathbb{F}_q$ with coefficients and a base point P of large order n is selected and made public to all users. The protocol considers a signature scheme *SIGN* = (*SIGN.Kgen, SIGN.Sig, SIGN.Ver*). Each client U_i holds a pair of signing private/public key $(\mathrm{SK}_i, \mathrm{PK}_i)$, which are the output of the key generation signature scheme algorithm *SIGN.Kgen*.

4.1. Key Generation. The algorithm *AGKA.Kgen*, on input of the set of clients ψ_C and a security parameter ℓ, performs the following steps.

(1) Execute *SIGN.Kgen*(ℓ) for each client U_i in ψ_C to provide each client with a pair $(\mathrm{SK}_i, \mathrm{PK}_i)$ of signing/verifying keys. The private key SK_i is given to the client U_i in a confidential way, while each public key PK_i is sent to the gateway.

(2) Choose random integer q_v, compute $Q_V = q_V * P$, and set the gateway's private/public keys $(\mathrm{SK}_V, \mathrm{PK}_V) = (q_V, Q_V)$. The private key is given to the gateway in a confidential way, while the public key is certified and sent to the clients. The pair (q_V, Q_V) will be the long-term Diffie-Hellman pair of the gateway.

Basically, for an ECC-based key agreement, each client will generate an ephemeral Diffie-Hellman pair (d_i, D_i), which thus leads to a session key R_i $(R_i = d_i * Q_V)$ shared between the client U_i and the gateway V. Meanwhile, ECDSA signature δ_i is used for authenticating each client node.

4.2. Group Key Setup. As depicted in Figure 2, the group key agreement setup runs as follows.

Step 1. To establish the group key in the cluster, each node $U_i \in \psi_C$ randomly selects a k-bit integer K_i and a $(160 - k)$-bit integer N_i as the nonce. Additionally, U_i randomly picks a random integer $d_i \in [2, n-2]$ as its ephemeral private key and gets the ephemeral public key $D_i = d_i * P$. Then, U_i computes $R_i = d_i * Q_V$ and cipher text $e_i = (K_i \Vert N_i) \oplus R_i \cdot x$, where $R_i \cdot x$ is the x coordinator of R_i. The client node U_i then generates an ECDSA signature $\delta_i = \mathrm{Sig}_{U_i}(D_i \Vert e_i)$ under the private key SK_i of U_i. Finally, each node U_i sends (D_i, e_i, δ_i) to gateway V. Note that generations of the ephemeral public key D_i and the shared secret R_i can be precomputed before the node joins the network, which requires additional memory space but speeds up the protocol's execution.

Step 2. For each node, the gateway first checks if the nonce N_i is fresh and then checks the signature δ_i to authenticate each node U_i. If the authentication holds, it computes $R_i = D_i * q_V$ and then decrypts e_i and gets K_i and N_i. Subsequently, the gateway initializes counter c and computes a group session key $\mathrm{GK} = H(c \Vert K_1 \Vert K_2 \Vert \cdots \Vert K_n)$. The gateway then creates a cipher text $e_{V_i} = \mathrm{GK} \oplus R_i \cdot x$ and sends each client node the cipher text e_{V_i}, counter c, and nonce N_i with $\mathrm{MAC}((e_{V_i} \Vert c), K_i)$. Note that secret key K_i is selected as the MAC key between node U_i and the gateway since K_i is only known to the node U_i and the gateway.

Step 3. Each sensor node U_i first performs the authentication of the gateway through verifying MAC. If the authentication holds, the client calculates the group session key $\mathrm{GK} = e_{V_i} \oplus R_i \cdot x$. HMAC with MD5 hash algorithm is used to calculate the MAC value. MAC is used to verify the integrity of the received message. MAC can also be used to confirm that the received message is sent by the sender who knows the MAC key K_i.

4.3. Algorithm for New Node Joining. The algorithm *AGKA.Join*, on input of the set of appearing client devices ϑ, performs the following steps.

(1) When a new member $U_{i+1} \in \vartheta$ wants to join a group, it must first be authenticated by the base station.

(2) Update the wireless client group $\psi_C = \psi_C \cup \vartheta$.

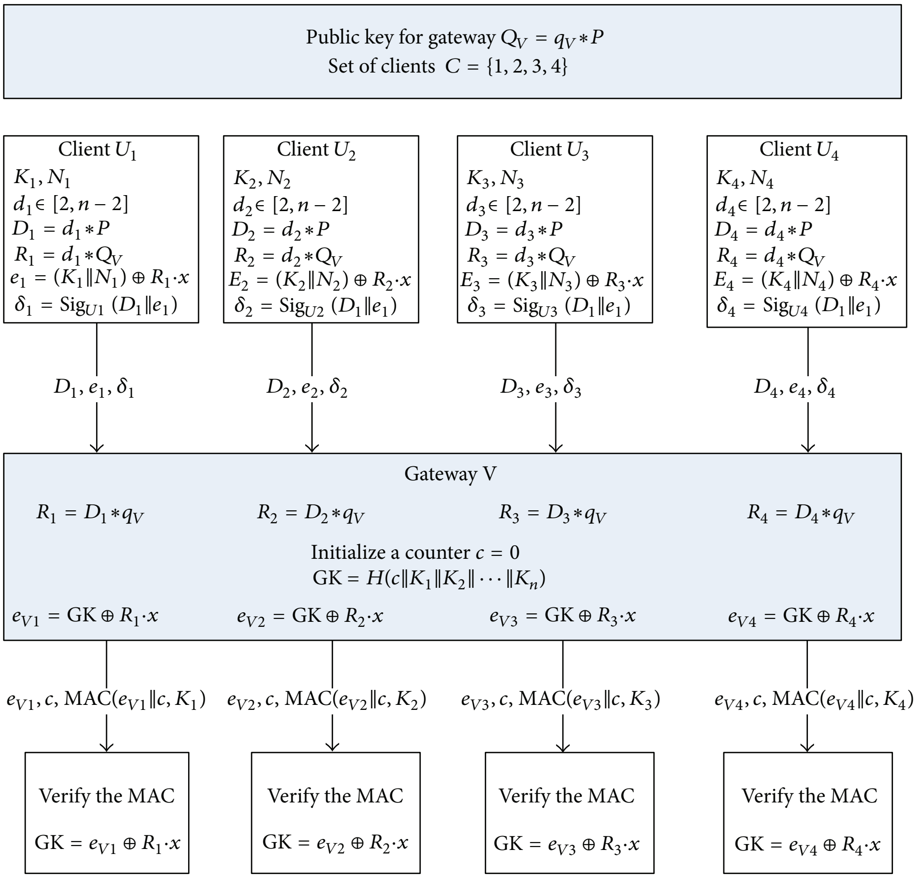

Public key for gateway $Q_V = q_V * P$
Set of clients $C = \{1, 2, 3, 4\}$
Client U_1
K_1, N_1
$d_1 \in [2, n-2]$
$D_1 = d_1 * P$
$R_1 = d_1 * Q_V$
$e_1 = (K_1 \| N_1) \oplus R_1 \cdot x$
$\delta_1 = \mathrm{Sig}_{U1}(D_1 \| e_1)$
Client U_2
K_2, N_2
$d_2 \in [2, n-2]$
$D_2 = d_2 * P$
$R_2 = d_2 * Q_V$
$E_2 = (K_2 \| N_2) \oplus R_2 \cdot x$
$\delta_2 = \mathrm{Sig}_{U2}(D_1 \| e_1)$
Client U_3
K_3, N_3
$d_3 \in [2, n-2]$
$D_3 = d_3 * P$
$R_3 = d_3 * Q_V$
$E_3 = (K_3 \| N_3) \oplus R_3 \cdot x$
$\delta_3 = \mathrm{Sig}_{U3}(D_1 \| e_1)$
Client U_4
K_4, N_4
$d_4 \in [2, n-2]$
$D_4 = d_4 * P$
$R_4 = d_4 * Q_V$
$E_4 = (K_4 \| N_4) \oplus R_4 \cdot x$
$\delta_4 = \mathrm{Sig}_{U4}(D_1 \| e_1)$
D_1, e_1, δ_1
D_2, e_2, δ_2
D_3, e_3, δ_3
D_4, e_4, δ_4
Gateway V
$R_1 = D_1 * q_V$
$R_2 = D_2 * q_V$
$R_3 = D_3 * q_V$
$R_4 = D_4 * q_V$
Initialize a counter $c = 0$
$\mathrm{GK} = H(c \| K_1 \| K_2 \| \cdots \| K_n)$
$e_{V1} = \mathrm{GK} \oplus R_1 \cdot x$
$e_{V2} = \mathrm{GK} \oplus R_2 \cdot x$
$e_{V3} = \mathrm{GK} \oplus R_3 \cdot x$
$e_{V4} = \mathrm{GK} \oplus R_4 \cdot x$
$e_{V1}, c, \mathrm{MAC}(e_{V1} \| c, K_1)$
$e_{V2}, c, \mathrm{MAC}(e_{V2} \| c, K_2)$
$e_{V3}, c, \mathrm{MAC}(e_{V3} \| c, K_3)$
$e_{V4}, c, \mathrm{MAC}(e_{V4} \| c, K_4)$
Verify the MAC
$\mathrm{GK} = e_{V1} \oplus R_1 \cdot x$
Verify the MAC
$\mathrm{GK} = e_{V2} \oplus R_2 \cdot x$
Verify the MAC
$\mathrm{GK} = e_{V3} \oplus R_3 \cdot x$
Verify the MAC
$\mathrm{GK} = e_{V4} \oplus R_4 \cdot x$

FIGURE 2: The AGKA protocol with five devices U_1, U_2, U_3, U_4, and V.

(3) Each appearing client $U_j \in \vartheta$ chooses at random a k-bit integer K_j, a $(160-k)$-bit integer N_j as the nonce, and the ephemeral private key $d_j \in [2, n-2]$ and precomputes the ephemeral public key $D_j = d_j * P$ and the shared session key $R_j = d_j * Q_V$. Then, U_j precomputes the cipher text $e_j = (K_j \| N_j) \oplus R_j \cdot x$ and the signature $\delta_j = \mathrm{Sig}_j(D_j \| e_j)$ under the private key SK_j.

(4) Each appearing client U_j sends the value (D_j, e_j, δ_j) to the gateway V.

(5) The gateway V verifies the incoming signatures and if correct, operates as in the *Setup* phase with an increased counter c and computes the group session key

$$\mathrm{GK} = H\left(c \,\|\, \left\{K_j\right\}_{j\in\vartheta}\right). \qquad (1)$$

After that, the gateway sends to each client $U_i \in \psi_C$ the counter c, cipher text $e_{V_i} = \mathrm{GK} \oplus R_i \cdot x$, and $\mathrm{MAC}((e_{V_i} \| c), K_i)$.

(6) Each client $U_i \in \psi_C$ already holds the value K_i, the shared secret R_i, and the old counter value. So, it first checks that the new counter is greater than the old one and the MAC value, and if the check holds, it simply recovers the group session key $\mathrm{GK} = e_{V_i} \oplus R_i \cdot x$.

4.4. Algorithm for Node Removing. The algorithm *AGKA.Remove*, on input of the set ϑ of disappearing client-sensors, performs the following steps.

(1) Update the sensor group $\psi_C = \psi_C / \vartheta$.

(2) The gateway V operates as in the *Setup* phase. It increases the counter c and computes the shared group session key $\mathrm{GK} = H(c \| \{K_i\}_{i\in\psi_C})$.

(3) Then, it sends to each client $U_i \in \psi_C$ the values c, cipher text $e_{V_i} = \mathrm{GK} \oplus R_i \cdot x$, and $\mathrm{MAC}((e_{V_i} \| c), K_i)$.

(4) Each client $U_i \in \psi_C$ already holds the value K_i, the shared secret R_i, and the old counter value. So, it first checks that the new counter is greater than the old one and the MAC value, and if the check holds, it simply recovers the group session key $\mathrm{GK} = e_{V_i} \oplus R_i \cdot x$.

5. Security Evaluation

The presented AGKA protocol overcomes the security weaknesses detected in the previously discussed protocols. The security evaluation is discussed in this section.

5.1. Sensor Node Replication Attack. The fresh nonce N_i is used in the message sent from the client node U_i for $i = 1, 2, \ldots n$, so that it can make sure no replayed message (cloning fraud) will be allowed in the protocol. For instance, if an adversary wants to replay the previously transmitted message from one client, it would use the same nonce value in previous round, which will be realized by the gateway who knows the last nonce generated by the client. If an adversary wants to replay the previously transmitted message from the gateway, it would not pass the check of the counter c implemented in Step 3 on the client side. Meanwhile, the signature of the message sent from the client node is also utilized in Step 1 to provide the authentication of the client nodes. Therefore, the proposed protocol prevents the replication attacks.

5.2. Sybil Attack. In this attack, a malicious sensor claims multiple IDs (identities) or locations [31]. In the proposed scheme, each client sensor is authenticated by the base station and gets a unique ID. In addition, each client owns a long-term key pair $(\mathrm{SK}_i, \mathrm{PK}_i)$, where the private key SK_i is used to generate the digital signature of the client. The private key is only known by the private key's owner and kept in secret. A malicious sensor cannot masquerade a forge ID and forge key pair without the base station's authentication. During the *AGKA.Setup* phase, the client's private key is used to sign the sending message, when the gateway in the group receives the signed message from a client node; it will first verify the signature $\delta_j = \mathrm{Sig}_j(D_j \parallel e_j)$ in order to authenticate the identity of the client node. Elliptic Curve Digital Signature Algorithm (ECDSA) is chosen in the proposed protocol to generate and verify the signature of each client. The security of ECDSA is founded in the difficulty of solving the discrete logarithm problem in prime order subgroups of $\mathbb{Z}_p^*$. The adversary cannot masquerade the client U_i and generate the legal signature to pass gateway's authentication without the private key of the client U_i. Even in worst case, the adversary compromise one client sensor N_i but still is not able to claim a new identity N_i' in the vicinity of node N_j because the adversary only knows the private key of the compromised node N_i but not the private key of node N_j. As a result, with the use of ECDSA on the gateway to authenticate the identity of each client sensor, the proposed protocol can withstand the Sybil attack.

5.3. Mutual Authentication. The signature of the message sent from the client node is generated in Step 1, which is verified by the gateway in Step 2. This provides the authentication of the client node. Meanwhile, a Message Authentication Code (MAC) is applied in Step 2. This will provide proof of authentication and integrity for the sent message. In the proposed protocol, the MAC key K_i is generated by client node U_i and sent to the gateway in a confidential way, where K_i is encrypted by $e_i = (K_i \parallel N_i) \oplus R_i \cdot x$. Only the gateway with the private key of q_V can decrypt the encrypted message and recover K_i. Thus, only the gateway V and client node U_i knows the MAC key K_i. Therefore, the MAC code $\mathrm{MAC}((e_{V_i} \parallel c), K_i)$ can be used to authenticate the identity of the gateway. As a result, the AGKA protocol provides the authentication between the client nodes and the gateway.

5.4. Perfect Forward Secrecy. A key agreement protocol offers forward secrecy if compromisation of a long-term key cannot result in the compromisation of previously established session keys. As mentioned in Step 1 of the AGKA protocol, $(d_i, D_i, R_i, e_i, \delta_i)$ is stored in the memory storage of the low-power node and each tuple $(d_i, D_i, R_i, e_i, \delta_i)$ is used only once. In this case, $(d_i, D_i, R_i, e_i, \delta_i)$ must be erased as soon as they are no longer useful. Obviously, since the low-power nodes' long-term keys SK_i are used only for authentication and they are not used for hiding the group key, the leakage of any client node's long-term key does not reveal anything about the group key. Furthermore, strong (partial) forward-secrecy (where any internal data is revealed, that is, the signing key but also the d_i, k_i, and R_i) is also achieved if the d_i's and R_i's are erased as soon as they are no longer useful (the client has left from the group). As a consequence, no information about previous session keys can be found in the memory of the low-power sensor nodes.

6. Formal Verification of the AGKA Protocol

Traditionally, cryptographic protocols have been designed and verified using informal and intuitive techniques. However, an absence of formal verification has proven [32, 33] to lead to flaws and security errors remaining undetected in a protocol. Formal verification aims at providing a rigid and thorough means of testing the correctness of a cryptographic protocol so that even subtle defects can be uncovered. A number of formal techniques have been developed for this purpose. This section first discusses the Coffey-Saidha-Newe (CSN) logical technique [32] and then formally analyzes and verifies the proposed group key agreement protocol using this logic.

6.1. CSN Modal Logic. The CSN logic provides a means of verifying hybrid cryptographic protocols. The logic can analyze the evolution of both knowledge and belief during a protocol execution, and is therefore useful in addressing issues of both security and trust. The inference rules provided are the standard inferences required for natural deduction and the axioms of the logic are sufficiently low-level to express the fundamental properties of hybrid cryptographic protocols, such as the ability of a principal to encrypt/decrypt based on knowledge of a cryptographic key. The logic is capable of analyzing a wide variety of hybrid cryptographic protocols because the constructs of the logic areof general purpose and therefore provide the user with increased flexibility allowing him to develop his own theorem.

The underlying assumptions of the logic can also be stated as follows. The communication environment is hostile but reliable; the cryptosystems used are ideal. That is, the encryption and decryption functions are completely noninvertible without knowledge of the appropriate cryptographic key and are invertible with knowledge of the appropriate cryptographic key. Keys used by the system are considered valid if they have not exceeded their validity period and only known by the rightful owner(s).

6.1.1. The CSN Logic Language

$a, b, c, \ldots$: general propositional variables

Φ: an arbitrary statement

Σ and Ψ: arbitrary entities

i and j: individual entities

ENT: the set of all possible entities

k: a cryptographic key. In particular, k_{Σ} is the public key of entity Σ and k_{Σ}^{-1} is the corresponding private key of entity Σ

$t, t', t'', \ldots$: moments in time. For example, $t1$ represents time after Step 1 of protocol has completed

$e(x, k_{\Sigma})$: encryption function, encryption of x using key k_{Σ}

$d(x, k_{\Sigma}^{-1})$: decryption function, decryption of x using key k_{Σ}^{-1}

$\text{ks}_{(\Sigma,\Psi)}$: shared secret key for entities Σ and Ψ

$\text{KS}_{\{\Sigma,\Psi\}}$: set of good shared keys for entities Σ and Ψ

$\text{ss}_{(\Sigma,\Psi)}$: shared secret for entities Σ and Ψ (secret can be fresh)

$\text{SS}_{\{\Sigma,\Psi\}}$: set of good shared secrets for entities Σ and Ψ

$E(x, \text{ks}_{(\Sigma,\Psi)})$: encryption of plaintext message x using the shared secret key of entities Σ and Ψ

$D(x, \text{ks}_{(\Sigma,\Psi)})$: decryption of ciphertext message x using the shared secret key of entities Σ and Ψ

K: propositional knowledge operator (true or false evaluation) of Hintikka [34]

$K_{\Sigma,t}\Phi$: Σ knows statement Φ at time t

L: knowledge predicate (assigns an object a property). $L_{\Sigma,t}x$ means that Σ knows and can reproduce object x at time t

B: belief operator. $B_{\Sigma,t}\Phi$ means that Σ believes at time t that statement Φ is true

C: "Contains" operator. $C(x, y)$ means that the object x contains the object y. The object y may be cleartext or ciphertext in x

S: emission operator. $S(\Sigma, t, x)$ means that Σ sends message x at time t

R: reception operator. $R(\Sigma, t, x)$ means that Σ receives message x at time t

A: authentication operator. $A(\Sigma, t, \Psi)$ means that Σ authenticates Ψ at time t.

The language includes the classical logical connectives of conjunction ($\wedge$), disjunction ($\vee$), complementation ($\neg$), and material implication ($\rightarrow$). The symbols $\forall$ and $\exists$ denote universal and existential quantification, respectively. The symbol $\in$ indicates membership of a set and / denotes set exclusion. The symbol $\vdash$ denotes a logical theorem. The logic does not contain specific temporal operators, but the knowledge, belief, and message transfer operators are time-indexed.

6.1.2. Inference Rule. The logic incorporates the following rules of inference.

(R1) From $\vdash p$ and $\vdash (p \rightarrow q)$ infer $\vdash q$.

(R2) (a) From $\vdash p$ infer $\vdash K_{\Sigma,t}p$;
(b) from $\vdash p$ infer $\vdash B_{\Sigma,t}p$.

(R1) is the *Modus Ponens* and states that if p can be deduced and $(p \rightarrow q)$ can be deduced, then q can also be deduced. (R2) consists of the generalisation rules which state that if p is a theorem, then knowledge and belief in p are also theorems.

The logic also includes the following standard propositional rules of natural deduction.

(R3) From $(p \wedge q)$ infer p.

(R4) From p and q infer $(p \wedge q)$.

6.1.3. Axioms. Two types of axioms are used in this logic, logical and nonlogical. Logical axioms are general statements made in relation to any system, while non-logical are system specific.

Logical Axioms. The logic includes the following standard modal axioms for knowledge and belief:

(A1) $\exists t \exists p \exists q (K_{\Sigma,t}p \wedge K_{\Sigma,t}(p \rightarrow q) \rightarrow K_{\Sigma,\mathrm{t}}q)$;

(A2) $\exists t \exists p (K_{\Sigma,t}p \rightarrow p)$.

The axiom (A1) is application of the Modus Ponens to the knowledge operator. The axiom (A2) is called the knowledge axiom and is said to logically characterise knowledge. If something is known, then it is true. This property distinguishes between knowledge and belief. Consider

(A3) (a) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(L_{i,t}x \rightarrow \forall t', t' \geq t\ L_{i,t}'x)$;
(b) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(K_{i,t}x \rightarrow \forall t', t' \geq t\ K_{i,t}'x)$.

Axioms (A3)(a) and (A3)(b) assert that knowledge, once gained, cannot be lost. Consider

(A4) $\exists t \exists x \exists y (\exists i, i \in \{\text{ENT}\} L_{i,t}y \wedge C(y, x) \rightarrow \exists j, j \in \{\text{ENT}\} L_{j,t}x)$.

If a piece of data is constructed from other pieces of data, then each piece of data involved in the construction must be known to some entity.

Nonlogical Axioms. The non-logical axioms reflect the underlying assumptions of the logic. These assumptions relate to the emission and reception of messages and to the use of encryption and decryption in these messages. Consider

(A5) $\exists t \exists x (S(\Sigma, t, x) \rightarrow L_{\Sigma,t}x \wedge \exists i, i \in \{\text{ENT}/\Sigma\} \exists t', t' > t\ R(i, t', x))$.

The emission axiom (A5) states that if Σ sends a message x at time t, then Σ knows x at time t and some entity i other than that Σ will receive x at time t' subsequent to t. Consider

(A6) $\exists t \exists x (R(\Sigma, t, x) \rightarrow L_{\Sigma,t}x \wedge \exists i, i \in \{\text{ENT}/\Sigma\} \exists t', t' < t\ S(i, t', x))$.

The reception axiom (A6) states that: if Σ receives a message x at time t, then Σ knows x at time t and some entity i other than that Σ has sent x at time t' prior to t. Consider

(A7) (a) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(L_{i,t}x \wedge L_{i,t}k_{\Sigma} \rightarrow L_{i,t}(e(x, k_{\Sigma})))$;

(b) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(L_{i,t}x \wedge L_{i,t}k_{\Sigma}^{1} \rightarrow L_{i,t}(d(x, k_{\Sigma}^{-1})))$.

Axioms (A7)(a) and (A7)(b) refer to the ability of an entity to encrypt or decrypt a message when it has knowledge of a public or private cryptographic key. Consider

(A8) (a) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(\neg L_{i,t}k_{\Sigma} \wedge \forall t', t' < t \neg L'_{i,t}(e(x, k_{\Sigma})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y, e(x, k_{\Sigma})))) \rightarrow \neg L_{i,t}(e(x, k_{\Sigma})))$;

(b) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(\neg L_{i,t}k_{\Sigma}^{-1} \wedge \forall t', t' < t \neg L'_{i,t}(d(x, k_{\Sigma}^{-1})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y, d(x, k_{\Sigma}^{-1})))) \rightarrow \neg L_{i,t}(d(x, k_{\Sigma}^{-1})))$.

Axioms (A8)(a) and (A8)(b) refer to the impossibility of encrypting or decrypting a message without knowledge of the correct key. Axiom (A8)(a) states that if an entity does not know k at t and does not know, prior to t, the encryption $e(x, k_{\Sigma})$ and also does not receive $e(x, k_{\Sigma})$ at t in a message, then the entity cannot know $e(x, k_{\Sigma})$ at time t. Axiom (A8)(b) makes a similar statement for the decryption of a message x without knowledge of the decryption key. Consider

(A9) $\forall t(\forall i, i \in \{\text{ENT}\}L_{i,t}k_i^{-1} \wedge \forall j, j \in \{\text{ENT}/i\}\neg L_{j,t}k_i^{-1})$.

The key secrecy axiom (A9) states that the private keys used by the system are known only to their rightful owners. Consider

(A10) $\exists t \exists x(\exists i, i \in \{\text{ENT}\}L_{i,t}d(x, k_{\Sigma}^{-1}) \rightarrow L_{\Sigma,t}x)$.

Axiom (A10) states that if an entity knows and can reproduce $d(x, k_{\Sigma}^{-1})$ and k_{Σ} at time t; then it knows and can reproduce x, and this implies that this entity knows at time t that Σ knows and can reproduce x prior to t. Consider

(A11) (a) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(L_{i,t}x \wedge L_{i,t}\text{ks}_{(\Sigma,\Psi)} \rightarrow L_{i,t}(E(x, \text{ks}_{(\Sigma,\Psi)})))$;

(b) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(L_{i,t}y \wedge C(y, E(x, \text{ks}_{(\Sigma,\Psi)})) \wedge L_{i,t}\text{ks}_{(\Sigma,\Psi)} \rightarrow L_{i,t}(D(x, \text{ks}_{(\Sigma,\Psi)})))$.

Axiom (A11) refers to the ability an entity has to encrypt or decrypt a message using a symmetric system when it has knowledge of a secret key. Consider

(A12) (a) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(\neg L_{i,t}\text{ks}_{(\Sigma,\Psi)} \wedge \forall t', t' < t, \neg L'_{i,t}(E(x, \text{ks}_{(\Sigma,\Psi)})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y, E(x, \text{ks}_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(E(x, \text{ks}_{(\Sigma,\Psi)})))$;

(b) $\exists t \exists x \exists i, i \in \{\text{ENT}\}(\neg L_{i,t}\text{ks}_{(\Sigma,\Psi)} \wedge \forall t', t' < t, \neg L'_{i,t}(D(x, \text{ks}_{(\Sigma,\Psi)})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y, D(x, \text{ks}_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(D(x, \text{ks}_{(\Sigma,\Psi)})))$.

Axiom (A12) refers to the inability of an entity to encrypt or decrypt data without knowledge of the appropriate shared secret key. Consider

(A13) $\forall t((\forall i, i \in \{\text{ENT}/\Sigma, \Psi\}\neg L_{i,t}\text{ks}_{(\Sigma,\Psi)} \wedge \exists j, j \in \{\Sigma, \Psi\}L_{j,t}\text{ks}_{(\Sigma,\Psi)}) \rightarrow \text{ks}_{(\Sigma,\Psi)} \in \{\text{KS}_{\{\Sigma,\Psi\}}\})$.

Axiom (A13) states that only the rightful owners of a shared secret key know that key; this implies that this key is a good key. Consider

(A14) $\forall t((\forall i, i \in \{\text{ENT}/\Sigma, \Psi\}\neg L_{i,t}\text{ss}_{(\Sigma,\Psi)} \wedge \exists j, j \in \{\Sigma, \Psi\}L_{j,t}\text{ss}_{(\Sigma,\Psi)}) \rightarrow \text{ss}_{(\Sigma,\Psi)} \in \{\text{SS}_{\{\Sigma,\Psi\}}\})$.

Axiom (A14) states that only the rightful owners of a shared secret know that secret; this implies that this is a good secret. Finally

(A15) (a) $\exists x \exists t(A(\Sigma, t, \Psi) \rightarrow (L_{\Sigma,t}\text{ss}_{(\Sigma,\Psi)} \wedge \text{ss}_{(\Sigma,\Psi)} \in \{\text{SS}_{\{\Sigma,\Psi\}}\} \wedge R(\Sigma, t, x) \wedge C(x, \text{ss}_{(\Sigma,\Psi)}) \wedge \forall t', t' < t, \neg S(\Sigma, t', x) \rightarrow K_{\Sigma,t}(S(\Psi, t', x))))$;

(b) $\exists x \exists t(A(\Sigma, t, \Psi) \rightarrow (L_{\Sigma,t}k_{\Psi} \wedge L_{\Sigma,t}x \wedge R(\Sigma, t, y) \wedge C(y, e(\mathrm{x}, k_{\Psi}^{-1}))) \rightarrow (\forall t', t' < t, K_{\Sigma,t}(S(\Psi, t', y))))$.

(A15)(a) states that if Σ knows a secret $\text{ss}_{(\Sigma,\Psi)}$ that it shares with Ψ (the secret can be fresh), and this secret is a good secret, and Σ receives a message containing $\text{ss}_{(\Sigma,\Psi)}$ at t that it did not send, then Σ knows that Ψ sent this message prior to t.

(A15)(b) states that if Σ knows the public key of Ψ ($k\Psi$) and message x, and if Σ receives a message y containing $e(x, k_{\Psi}^{-1})$, then Σ knows that Ψ sent message y prior to t.

6.2. Formal Verification of the Proposed Protocol. To provide assurance that the new AGKA protocol is verifiably secure and trustworthy, a formal verification on its specifications is performed in this section. CSN logic was adopted to perform formal verifications of security protocols in Chapter 6, and is therefore adopted here to perform the formal verification of the new proposed group key agreement protocol.

6.2.1. Goals of the Proposed AGKA Protocol. The goals of the key-agreement protocol are defined as follows:

Goal 1: $K_{V,t1}$ $(\exists t, t < t1, S(U_i, t, X) \wedge C(X, (D_i, K_i \parallel N_i)))$, for $i = 1, \ldots, n$;

Goal 2: $K_{Ui,t2}$ $(\exists t, t1 < t < t2, S(V, t, X) \wedge C(X, (\text{GK}, c, N_i)))$, for $i = 1, \ldots, n$.

Goal 1 states that the gateway V knows that it will obtain a signed message from U_i containing the ephemeral public key D_i and the concatenation value $K_i \parallel N_i$ prior to the end of Step 1.

Goal 2 states that the low power node U_i will obtain a message from V containing the group key GK, the counter c, and the nonce N_i after Step 1 but before the end of Step 2.

6.2.2. Initial Assumptions. Consider the following:

(1) $\forall i, \forall t, i \in \{\text{ENT}\}(L_{i,t}Q_V \wedge L_{i,t}Q_i)$;

(2) $\forall i, \forall j, \forall t, i \in \{\text{ENT}/V\}\neg L_{i,t}q_V \wedge j \in \{\text{ENT}/U_i\}\neg L_{j,t}q_i$;

(3) $K_{Ui,t0}$ $(\forall j, \forall t, j \in \{\text{ENT}/U_i\}, t < t1, \neg L_{j,t}N_i)$;

(4) $K_{Ui,t0}$ $(\forall j, \forall t, j \in \{\text{ENT}/U_i\}, t < t1, \neg L_{j,t} R_i \cdot x)$;

(5) $\forall i, \forall j, \forall t, ((t > t1\ i \in \{U_i, V\} L_{i,t} R_i \cdot x) \wedge (j \in \{\text{ENT}/U_i, V\} \neg L_{j,t} R_i \cdot x)) \rightarrow (R_i \cdot x \in \text{SS}_{\{U_i,V\}})$;

(6) $L_{Ui,t0} K_i \wedge K_{Ui,t0} (\forall i, \forall t, i \in \{\text{ENT}/U_i\}, t < t1, \neg L_{j,t} K_i) \rightarrow (K_i \in \text{KS}_{\{Ui,V\}})$.

Assumption (1) states that the public keys Q_V and Q_i, where $i = 1 \ldots n$, are known to all entities.

Assumption (2) states that the private keys of U_i and V are known only to its owner and not known to any other entity.

Assumption (3) refers to the timely revelation of the random nonce N_i by the client U_i.

Assumption (4) refers to the timely revelation of the shared key $R_i \cdot x$ by the client U_i.

Assumption (5) states that only the entities U_i and V will know the shared key $R_i \cdot x$ after Step 1, and this implies that $R_i \cdot x$ is a good secret.

Assumption (6) states that U_i generates the shared MAC key K_i and that U_i knows that no other entity knows this key prior to $t1$, and that the key is a good key.

6.2.3. Formal Analysis

Step 1. $K_{V,t1}(R(V, t1, X) \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x), e(\text{Mes}, q_i)))$ where $\text{Mes} = (D_i \parallel E(K_i \parallel N_i, R_i \cdot x))$.

This states that V knows at time $t1$, it will receive a message X containing the ephemeral public key D_i and encrypted message $E(K_i \parallel N_i, R_i \cdot x)$. And this message will be signed by the private key of the client.

By application of Axiom (A2),

$$R(V, t1, X) \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x), e(\text{Mes}, q_i))). \tag{2}$$

Applying Axiom (A6) and Inference Rule (R2),

$$\begin{gathered} L_{V,t1} X \wedge K_{V,t1} \left(\exists j, j \in \left\{\frac{\text{ENT}}{V}\right\}\right), \\ \exists t, t < t1, S(j, t, X) \\ \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x), e(\text{Mes}, q_i))). \end{gathered} \tag{3}$$

Applying Inference Rule (R3),

$$\begin{gathered} K_{V,t1} \left(\exists j, j \in \left\{\frac{ENT}{V}\right\}\right), \\ \exists t, t < t1, S(j, t, X) \\ \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x), e(\text{Mes}, q_i))). \end{gathered} \tag{4}$$

Using Assumption (4) which states that only U_i has knowledge of $R_i \cdot x$ before $t1$ and Assumption (3) which states that only U_i has knowledge of N_i before $t1$,

$$\begin{gathered} K_{V,t1} (\exists t, t < t1), \\ S(U_i, t, X) \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x), e(\text{Mes}, q_i))). \end{gathered} \tag{5}$$

Using Axioms (A7)/(A8)/(A15)(b) which reflect the ability of an entity to authenticate another entity when it has knowledge of its public key and a message with a signature of the message, Assumption (1) which states that the public key of U_i is known to all entities, and Assumption (2) which states the private key of U_i is only known to its owner U_i, we get,

$$\begin{gathered} A(V, t1, U_i) \longrightarrow K_{V,t1} (\exists t, t < t1), \\ S(U_i, t, X) \wedge C(X, (D_i, E(K_i \parallel N_i, R_i \cdot x))). \end{gathered} \tag{6}$$

This shows that the client U_i is authenticated at Step 1 of the protocol since only it could have encrypted *Mes* with its secret key q_i and *Mes* contains the ephemeral public key D_i, and the cipher text $E(K_i \parallel N_i, R_i \cdot x)$.

Using Axioms (A11) and (A12), which reflect the ability of an entity to decrypt a message when it has knowledge of the secret key, and Assumption (5) which states that $R_i \cdot x$ is a good secret key only known to U_i, and V we get,

$$\begin{gathered} K_{V,t1} (\exists t, t < t1, S(U_i, t, X) \wedge C(X, (D_i, K_i \parallel N_i))), \\ \text{: satisfying Goal 1.} \end{gathered} \tag{7}$$

Step 2.

$$\begin{gathered} K_{Ui,t2} (R(U_i, t2, X)) \\ \wedge C(X, (E(\text{GK}, R_i \cdot x), c, N_i, \text{MAC}(\text{Mes2}, K_i))), \end{gathered} \tag{8}$$

where $\text{Mes2} = E(\text{GK}, R_i \cdot x) \| c \| N_i$.

This states that U_i knows at time $t2$ that it will receive a message X containing cipher text $E(\text{GK}, R_i \cdot x)$, nonce N_i, and a message authentication code of this message.

By application of Axiom (A2),

$$\begin{gathered} R(U_i, t2, X) \\ \wedge C(X, (E(\text{GK}, R_i \cdot x), c, N_i, \text{MAC}(\text{Mes2}, K_i))). \end{gathered} \tag{9}$$

Applying Axiom (A6) and Inference Rule (R2),

$$\begin{gathered} L_{Ui,t2} X \wedge K_{Ui,t2} (\exists j, j \in \{\text{ENT}/U_i\}, \exists t, t < t2), \\ S(j, t2, X) \wedge C(X, (E(\text{GK}, R_i \cdot x), c, N_i, \text{MAC}(\text{Mes2}, K_i))). \end{gathered} \tag{10}$$

Applying Inference Rule (R3)

$$\begin{gathered} K_{Ui,t2} (\exists j, j \in \{\text{ENT}/U_i\}, \exists t, t < t2), \\ S(j, t2, X) \wedge C(X, (E(\text{GK}, R_i \cdot x), c, N_i, \text{MAC}(\text{Mes2}, K_i))). \end{gathered} \tag{11}$$

Applying Axioms (A11)/(A12) and (A13) and Assumption (4), and using Assumption (5) which states that $R_i \cdot x$ is a good secret key only known to entities U_i and V:

$$\begin{gathered} K_{Ui,t2} (\exists t, t < t2), \\ S(V, t2, X) \wedge C(X, (E(\text{GK}, R_i \cdot x), c, N_i, \text{MAC}(\text{Mes2}, K_i))). \end{gathered} \tag{12}$$

Using Assumption (3) which states the timely revelation of N_i (after time $t1$) by U_i, we get

$$K_{Ui,t2}\left(\exists t, t1 < t < t2\right),$$
$$S\left(V, t2, X\right) \wedge C\left(X, \left(E\left(\text{GK}, R_i \cdot x\right), c, N_i, \text{MAC}\left(\text{Mes2}, K_i\right)\right)\right). \tag{13}$$

The client V is authenticated at this point of the protocol since only U_i and V could have encrypted *Mes2* and generate the Message Authentication Code MAC(Mes2, K_i) with its secret key K_i (Axioms (A11)/(A12)/(A15)(a) and Assumption (6)), and *Mes2* contains the cipher text $E(\text{GK}, R_i \cdot x)$, the nonce N_i, and the counter c; therefore

$$A\left(U_i, t1, V\right) \longrightarrow K_{Ui,t2}\left(\exists t, t1 < t < t2\right),$$
$$S\left(V, t2, X\right) \wedge C\left(X, \left(E\left(\text{GK}, R_i \cdot x\right), c, N_i\right)\right). \tag{14}$$

Using Axioms (A11) and (A12), which reflect the ability of an entity to decrypt a message when it has knowledge of the secret key, and Assumption (5) which states that $R_i \cdot x$ is a good secret key only known to U_i and V, we get

$$K_{Ui,t2}\left(\exists t, t1 < t < t2, S\left(V, t2, X\right) \wedge C\left(X, \left(\text{GK}, c, N_i\right)\right)\right),$$
$$\text{: satisfying Goal 2.} \tag{15}$$

From the analysis it can be seen that all goals of the proposed group key agreement protocol are achieved and no security flaw is detected. This indicates that the proposed protocol is verifiably secure and trustworthy.

7. Implementation and Performance Evaluation

In order to evaluate the suitability of our protocol in sensor networks, we carried out a set of experiments based on the TelosB [35] and MICAz [36] mote platforms. Table 2 lists the configuration and the architecture of TelosB and MICAz motes.

A low-end PC (1.0 GHz Intel Pentium III processor, 512 MB RAM, and 30 GB hard drive) with a mote attached is used to simulate the gateway. The TelosB mote or the MICAz mote attached to the PC is responsible for transmitting and receiving messages. Using the PC as the security manager enables the security manager to implement all operations by the Java program and store all members' public keys in the local memory device without worrying about memory constraints. This method reduces the execution time of the protocol and releases the memory and power constraints existing in sensor nodes. Most cryptographic algorithms, such as ECDSA, RC5, and Skipjack, are supported by Java, and these algorithms can be found in the Java security packages or the third-party security packages. Another reason for using the PC to simulate the gateway is that the handshaking messages and execution process can be displayed on PC, which eases the researchers in tracing the messages received from the group members and the authentication process during the AGKA protocol.

7.1. Implementation. The implementation is divided into two modules, the client (group member) module and the security manager module.

(i) The client module implements all the operations required by the proposed protocol on the client side, which involves ECC point multiplication, ECDSA signature generation, and MAC generation.

(ii) The security manager module has two parts. The first part powernode.nc is written in nesC code and implemented on the MICAz and TelosB that are attached to the security manager (computer), and the other part is securitymanger.java which is written in Java and implemented on the security manager (computer). These two parts are linked by a Java class MoteIF which enables Java applications to send and receive the message through Universal Asynchronous Receiver/Transmitter (UART).

In software, we implemented our protocol by the use of the nesC programming language and work with the TinySec [37] module and the TinyECC [23] software package, implemented specifically for TinyOS.

TinySec is the first fully implemented link layer security architecture for wireless sensor networks. It is also a research platform that is easily extendable and has been incorporated into higher level protocols. Some well-studied cryptographic primitives are applied in TinySec, such as Message Authentication Codes (MACs), Initialization Vectors (IVs), and Cipher Block Chaining (CBC). It is noteworthy that TinySec was distributed with official releases of TinyOS version 1.x. It has proven that efficient secure communication in wireless sensor networks is a feasible reality. Table 3 summarizes the security characteristics of TinySec.

The TinyECC package supports all elliptic curve operations over prime fields F_p, including point addition, point doubling, and scalar point multiplication, as well as ECDSA operations. It also includes elliptic curve parameters recommended by Stands for Efficient Cryptography Group (SECG), such as secp160k1, secp160r1, and secp160r2. The natural number operations in TinyECC are based on RSAREF2.0 [23, 38].

Bouncy Castle [39] is a collection of APIs used in cryptography. It includes APIs for both the Java and the C# programming languages. It provides a Java library to implement all elliptic curve operations over F_p, including point addition, point doubling, and scalar point multiplication, as well as ECDSA operations. In order to implement ECDSA operations in Java, a number of Bouncy Castle classes are imported into our implementation.

7.2. Experimental Setup. The performance evaluation is performed on both TelosB and MICAz motes. We set two experimental networks, both consist of groups of seven client motes and a single gateway. The performance of the protocol in each network is evaluated. As mentioned in Section 4, some values such as D_i and R_i can be pre-computed before the sensor node *AKGA.SETUP* phase. This is to facilitate a

TABLE 2: Configuration of TelosB and MICAz motes.

Mote	Manufacturer	Microcontroller	Clock frequency	RAM	Program memory	Data memory	Radio
MICAz	Crossbow	Atmega 128	7.37 MHz	4 kB	128 kB	512 kB	CC2420
TelosB	Moteiv	TI MSP430	4 MHz	10 kB	48 kB	1 MB	CC2420

TABLE 3: TinySec security characteristics.

	Encryption	Block cipher	Code requirement	Auth. provided	Cost (time/energy)	Key agreement
TinySec	Optional—CBC mode (with CTS)	Skipjack/RC5	7146 Bytes Max.	Yes—CBC-MAC	0.38 ms/9.1% Max.	No

speeding up of the protocol's operation. The impact of the use of precomputation methods will be evaluated.

To enable TelosB and MICAz motes to execute the ECC computations required by the AGKA protocol, the 128-bit and 160-bit ECC parameters recommended by SECG [40] are chosen for use in the tests presented in the experiment, while the 192-bit ECC parameters are not included in the evaluation. This is because the 192-bit ECC requires 48 bytes to represent the point (public key pair) on the curve, which results in 120 bytes payload in the communication message; such large payload size exceeds the maximum TinyOS payload size of 114 bytes.

The following evaluating measurements are used in our performance evaluation experiments:

(i) ROM consumption;

(ii) RAM consumption;

(iii) execution time;

(iv) energy consumption.

7.3. Evaluation Results. A comparison between the results on the TelosB and the results on the MICAz, as well as between the results with pre-computation disabled and with pre-computation enabled, will now be presented.

7.3.1. Execution Time. The execution time can be one of the most meaningful attributes when evaluating security protocols, especially with regard to resource-constrained sensor nodes. The execution time is measured using an oscilloscope.

In comparing two different mote architectures with the same protocol running, it can be seen that the resulting execution time depends on the clock frequency of the microcontroller on the sensor platform.

Figure 3 plots the average execution times for the AGKA protocol implemented on both the TelosB and the MICAz motes with different elliptic curves.

From Figure 3, it can be seen that the value for the execution time on the MICAz mote is about half that of the TelosB mote results, and this can be attributed to the clock frequency of the MICAz being 8 MHz which is double the clock frequency of the TelosB mote. Different elliptic curves affect the execution time of the protocol, and this can be seen in the fact that there is at least a 1.00 second difference with 128-bit elliptic curves implemented compared with 160-bit elliptic curves. It is noticeable that the execution time is significantly reduced when pre-computation is enabled; the reason for this is that two public-key generations are pre-computed and the corresponding results are installed in the memory before the nodes join the network. This saves at least 9 seconds in execution time for the TelosB mote and saves at least 4.50 seconds in execution time for the MICAz mote. The fastest execution time observed from the experimental results is 2.64 seconds, when the AGKA protocol with the secp128k1 elliptic curve was implemented on the MICAz motes. Although pre-computation speeds up the protocol, considerable increases in ROM usage are traded.

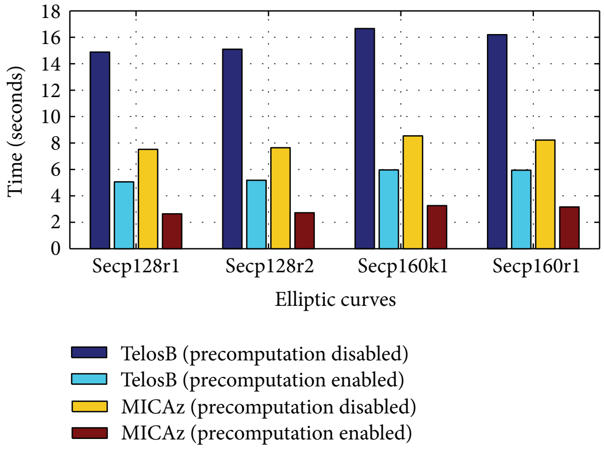


FIGURE 3: Comparison of the execution time on TelosB and MICAz motes.

7.3.2. Memory Usage. Due to the limited storage available on the sensor nodes, memory usage is an important attribute when evaluating the new key agreement protocol. As already mentioned, the pre-computation method improves the execution speed of the protocol; however, extra memory required is the tradeoff. The check_size script provided by the TinyOS is used to obtain the ROM and RAM sizes required by the AGKA protocol in each experiment.

The experiment evaluates the increases in ROM requirements of the proposed AGKA protocol with pre-computation enabled. Table 4 illustrates the ROM consumption for the

Table 4: ROM usage for the AGKA protocol on the TelosB and MICAz motes.

Elliptic curves	First run		Second run		Third run		Fourth run	
	ROM (bytes)	RAM (bytes)	ROM (bytes)	RAM (bytes)	ROM (bytes)	RAM (bytes)	ROM (bytes)	RAM (bytes)
Secp128r1	27716	2492	27938	2560	28216	2628	28514	2702
Secp128r2	27684	2492	27962	2560	28228	2628	28636	2702
Secp160k1	28876	2868	29214	2952	29536	3036	29874	3110
Secp160r1	28844	2868	29182	2952	29516	3036	29842	3110

AGKA protocol on the TelosB and MICAz motes when the pre-computation method is enabled.

It can be seen that the ROM consumption increases with a rise in the number of *AGKA.Setup* algorithms run. The reason for that is discussed in the following. In Step 1, each low-power node U_i uses the offline pre-computing technique to compute $D_i = d_i * p$, $R_i = d_i * Q_V$, $e_i = (K_i \| N_i) \oplus R_i \cdot x$ and a signature $\delta_i = \mathrm{Sig}_{U_i}(D_i \| e_i)$. Certainly, some tuples $(d_i, D_i, R_i, e_i, \delta_i)$ should be stored in the memory storage of the low-power node U_i in advance. When the proposed protocol plans to run four *AGKA.Setup* algorithms, it will store 4 tuples $(d_i, D_i, R_i, e_i, \delta_i)$ in the memory at beginning and give each tuple a sequence number; for example, the tuple 1 is named as $(d_i, D_i, R_i, e_i, \delta_i)^1$ and tuple 2 is named as $(d_i, D_i, R_i, e_i, \delta_i)^2$. This is the reason why the ROM consumption increases with a rise in the number of *AGKA.Setup* algorithms run. After each run, the proposed protocol will remove the corresponding used tuple $(d_i, D_i, R_i, e_i, \delta_i)$; for example, the protocol will remove the tuple 1 $(d_i, D_i, R_i, e_i, \delta_i)^1$ at the end of the first execution of the *AGKA.Setup* algorithm.

7.3.3. Energy Consumption. Another important evaluation measurement besides the memory usage and the execution time is the energy consumption. The energy consumption by the AGKA protocol is measured by the using of the Agilent mobile communication DC Source (DCS). Figure 4 illustrates the energy consumption for the AGKA protocol implemented on the TelosB and the MICAz motes with specific elliptic curves.

It is shown that the protocol with 128-bit elliptic curves consumes less energy than with 168-bit elliptic curves. This is attributed to a reduction in computational complexity and shorter message size when the protocol uses the 128-bit elliptic curves. With the same elliptic curve, the energy consumed by the protocol on the MICAz is less than that on the TelosB. The reason for this is that the execution times on the MICAz are about half that on the TelosB. Furthermore, with the same elliptic curve, at least 35 μWH of energy is saved with pre-computation enabled on the MICAz mote, while at least 32 μWH of energy is saved with pre-computation enabled on the TelosB mote.

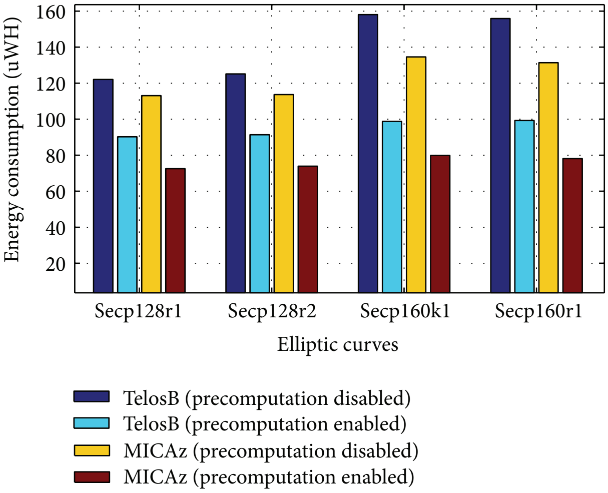


Figure 4: Comparison of energy consumption on TelosB and MICAz motes.

7.4. Limitation and Further Improvement. The comparison results identify that execution time and energy consumption are reduced with short elliptic curves, and those measurements are also improved with pre-computation enabled, while the significant increases in memory usage is the critical tradeoff. Therefore, further improvements and optimizations on memory usage need to be implemented in future work.

The experiment only evaluates the protocol with a group size of seven. With increasing the group size, the execution time will increase. The major reason is that the clients' handshaking packets will queue in the transceiver of the security manager and may cause the jam in the communication channel. Further experiments and simulations on protocol performance versus group size should be carried out.

8. Conclusion and Future Work

In this paper, a secure authenticated group key agreement protocol well suited for wireless sensor networks has been proposed. We showed that the proposed protocol provides forward secrecy and mutual authentication between low-power nodes and the powerful node (gateway). We also demonstrated that the proposed protocol is verifiably secure against node replication attacks and Sybil attacks. Meanwhile, the implementation of the protocol on the TelosB and the MICAz motes was also described in detail. In addition to the implementation of the protocol, a number of evaluation experiments were developed and performed on the motes and described. The experimental results were analyzed based on the following evaluation metrics: execution time, memory usage, and energy consumption. The evaluation results indicate that the protocol is suitable for use with energy-constrained sensor networks. We plan to further investigate the reduction method that can be used to reduce the bit-length of the pre-computed key pairs and signatures, which will in turn reduce the memory usage of the proposed protocol. In addition, we plan to carry out a further evaluation of the proposed protocol with a larger number of group members than used in this study.

Acknowledgments

This research work was supported by the National Natural Science Foundation of China (61103238 and 61003278). This research is also supported by the Fundamental Research Funds for the Central Universities.

References

[1] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November 2002.

[2] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE Symposium on Security And Privacy*, pp. 197–213, May 2003.

[3] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 52–61, October 2003.

[4] E. Bresson, O. Chevassut, A. Essiari, and D. Pointcheval, "Mutual authentication and group key agreement for low-power mobile devices," *Computer Communications*, vol. 27, no. 17, pp. 1730–1737, 2004.

[5] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Zhang, "Fast authenticated key establishment protocols for self-organizing sensor networks," in *Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '03)*, pp. 141–150, September 2003.

[6] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Location-based compromise-tolerant security mechanisms for wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 247–260, 2006.

[7] Y. H. Kim, H. Lee, J. H. Park, L. T. Yang, and D. H. Lee, "Key establishment scheme for sensor networks with low communication cost," in *Proceedings of the 4th International Conference on Autonomic and Trusted Computing: Bringing Safe, Self-x and Organic Computing Systems into Reality (ATC '07)*, vol. 4610, pp. 441–448, Hong Kong, Hong Kong, 2007.

[8] L. P. Zhang and Y. Wang, "An ID-based authenticated key agreement protocol for wireless sensor networks," *Journal of Communications*, vol. 5, no. 8, pp. 620–626, 2010.

[9] H. L. Yeh, T. H. Chen, P. C. Liu, T. H. Kim, and H. W. Wei, "A secured authentication protocol for wireless sensor networks using Elliptic Curves Cryptography," *Sensors*, vol. 11, no. 5, pp. 4767–4779, 2011.

[10] K. Ammayappan, A. Negi, V. N. Sastry, and A. K. Das, "An ECC-based two-party authenticated key agreement protocol for mobile Ad Hoc networks," *Journal of Computers*, vol. 11, pp. 2408–2416, 2011.

[11] H. F. Huang, "A new design of access control in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 412146, 7 pages, 2011.

[12] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[13] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 62–72, Washington, DC, USA, October 2003.

[14] J. Nam, S. Kim, and D. Won, "A weakness in the Bresson-Chevassut-Essiari-Pointcheval's group key agreement scheme for low-power mobile devices," *IEEE Communications Letters*, vol. 9, no. 5, pp. 429–431, 2005.

[15] J. Nam, J. Lee, S. Kim, and D. Won, "DDH-based group key agreement in a mobile environment," *Journal of Systems and Software*, vol. 78, no. 1, pp. 73–83, 2005.

[16] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," *Journal of Cryptology*, vol. 20, no. 1, pp. 85–113, 2007.

[17] Y. M. Tseng, "A secure authenticated group key agreement protocol for resource-limited mobile devices," *Computer Journal*, vol. 50, no. 1, pp. 41–52, 2007.

[18] A. Shamir and Y. Tauman, "Improved on-line/off-line signature schemes," in *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, pp. 355–367, Springer, Berlin, Germany, 2001.

[19] Y. Li and T. Newe, "On the logical verification of a group key agreement protocol for resource constrained mobile devices," in *Proceedings of the Australasian Telecommunication Networks and Applications Conference (ATNAC '07)*, pp. 277–281, Christchurch, New Zealand, December 2007.

[20] R. Struik and G. Rasor, *Mandatory ECC Security Algorithm Suite*, IEEE P802.15 Wireless Personal Area Networks, 2002.

[21] SECG, *SEC1: Elliptic Curve Cryptography, Standards For Efficient Cryptography Group*, Certicom Research, 2000.

[22] M. Aydos, T. Yan, and C. K. Koc, "A High-speed ECC-based wireless authentication protocol on an ARM microprocessor," in *Proceedings of the 16th Annual Computer Security Applications Conference (ACSAC '00)*, New Orleans, La, USA, 2000.

[23] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 245–256, April 2008.

[24] U. Berkeley, *TinyOS Community Forum: An Open Source OS for the Networked Sensor Regime*, 2007.

[25] C. P. Schnorr, "Efficient signature generation by smart cards," in *Proceedings of the 9th Annual International Cryptology Conference (CRYPTO '89)*, vol. 434 of *Lecture Notes in Computer Science*, pp. 688–689, Santa Barbara, Calif, USA, January 1991.

[26] Y. Zhang, W. Liu, Y. Fang, and D. Wu, "Secure localization and authentication in ultra-wideband sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. 829–835, 2006.

[27] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Securing sensor networks with location-based keys," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '05)*, pp. 1909–1914, March 2005.

[28] *IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks- Specific Requirements-Part 15.4: Wireless*

Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs), 2003.

[29] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[30] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *International Journal of Information Security*, vol. 1, pp. 36–63, 2001.

[31] J. Douceur, "The sybil attack," in *Procceeding of the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02)*, 2002.

[32] T. Coffey and P. Saidha, "Logic for verifying public-key cryptographic protocols," *IEE Computers and Digital Techniques*, vol. 144, pp. 28–32, 1997.

[33] M. Burrows, M. Abadi, and R. Needham, "Logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.

[34] J. Hintikka, *Knowledge and Belief: An Introduction to the Logic of Two Notions*, Cornell University Press, Ithaca, NY, USA, 1962.

[35] Crossbow. TelosB product http://www.xbow.com/Products/productsdetails.aspx?sid=147>.

[36] Crossbow-Technology. *MICAz Datasheet*, 2008, http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf.

[37] C. Karlof, N. Sastry, and D. Wagner, "TinySec: a link layer security architecture for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 162–175, Baltimore, Md, USA, November 2004.

[38] *RSA Laboratories, RSAREF: A cryptographic toolkit (version 2.0)*, 1994.

[39] Bouncy Castle, http://www.bouncycastle.org/.

[40] CC2431. Texas Instruments Incorporated, 2008, http://focus.ti.com/docs/prod/folders/print/cc2431.html.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 258081, 12 pages
http://dx.doi.org/10.1155/2013/258081

Research Article

Constructing a CDS-Based Network Backbone for Data Collection in Wireless Sensor Networks

Xiaoyan Kui,[1] Yu Sheng,[1] Huakun Du,[2] and Junbin Liang[3]

[1] *School of Information Science and Engineering, Central South University, Changsha 410083, Hunan, China*
[2] *School of Geoscience and Info-Physics, Central South University, Changsha 410083, Hunan, China*
[3] *School of Computer and Electronic Information, Guangxi University, Nanning 530004, Guangxi, China*

Correspondence should be addressed to Yu Sheng; shengyu@csu.edu.cn

Received 2 January 2013; Accepted 19 March 2013

Academic Editor: Mianxiong Dong

Data collection is one of the most important operations in wireless sensor networks. Currently, many researches focus on using a connected dominating set to construct a virtual backbone for data collection in WSNs. Most researchers concentrate on how to construct a minimum connected dominating set because a small virtual backbone incurs less maintenance. Unfortunately, computing a minimum size CDS is NP-hard, and the minimum connected dominating sets may result in unbalanced energy consumption among nodes. In this paper, we investigate the problem of constructing an energy-balanced CDS to effectively preserve the energy of nodes in order to extend the network lifetime in data collection. An energy-balanced connected dominating set scheme named DGA-EBCDS is proposed, and each node in the network can effectively transmit its data to the sink through the virtual backbone. When constructing the virtual backbone in DGA-EBCDS, we prioritize selecting those nodes with higher energy and larger degree. This method makes the energy consumption among nodes more balanced. Furthermore, the routing decision in DGA-EBCDS considers both the path length and the remaining energy of nodes in the path; it further prolongs the lifetime of nodes in the backbone. Our conclusions are verified by extensive simulation results.

1. Introduction

Wireless sensor network (WSN) is a new hot spot in current research, and it has broad application foreground [1, 2]. A wireless sensor network consists of a large number of nodes that collaborate together to monitor various phenomena. WSN can be used in various fields, such as battlefield, environment monitoring, disaster forecasting, business, and traffic control.

However, the communication ability, computational ability, storage ability, and energy of sensor nodes are limited [3–6], and the nodes are difficult to be replaced because they are often deployed in remote or inaccessible environments; the limitations of wireless sensor networks also shorten the performance of the network. Because of the limited energy, the signal generated by a source can only reach the nodes that are within its transmission range. If two nodes are within the transmission range of each other, they can exchange messages directly; otherwise, the messages have to be relayed through other intermediate nodes [7, 8]. This problem motivates us to construct a virtual backbone in wireless sensor network, and through this virtual backbone each node of the network can effectively transmit its data to the sink.

Wireless sensor networks (WSNs) are data oriented and are usually densely deployed in a monitor environment to process a great deal of data. Data collection is one of the most important operations in wireless sensor networks [9–13], which means that the data sensed by nodes should be transmitted to the sink for further processing. How to conserve the limited energy of nodes and extend the lifetime of the network is an important issue in data collection [14, 15]. The network lifetime is usually defined as the duration of the network until the first node depletes its energy. So, the network lifetime effectively ends with the first node death (FND). Due to incomplete data, the remaining energy in the surviving nodes is of no use after FND. In order to effectively extend the lifetime of a WSN (delay the death of the first node), many algorithms construct a virtual backbone of the network and only use the nodes in the virtual backbone to receive and transmit data for data collection. Nodes that are not in the backbone can go to sleep mode periodically to conserve the limited energy. Because of the number of nodes in the virtual backbone is relatively small, the impact of the broadcast storm problem can be greatly reduced, and the routing path search space can also be limited to the nodes that are in the set of the backbone. So, the virtual backbone can bring several benefits to network management.

Connected dominating set (CDS) plays an important role in the construction of virtual backbone in wireless sensor networks [16]. A dominating set (DS) of a network is a subset of all its nodes, which makes every node out of the set adjacent to at least one node in the set. A dominating set is called a connected dominating set when the nodes in the dominating set can form a connected graph. A CDS can be used as a virtual backbone to help each node transfer its data to the sink. Many researches focus on finding the minimum connected dominating set (MCDS) to construct the network virtual backbone [10]. However, MCDS-based algorithms may result in too much energy consumption of nodes in the backbone, and may cause unbalanced energy consumption among all the nodes, which will shorten the network lifetime and narrow their applications.

Motivated by the above problem, an energy-balanced connected dominating set scheme (DGA-EBCDS) is proposed in this paper, which well balances energy consumption among nodes that are in the backbone and consequently prolongs the lifetime of the network. In DGA-EBCDS, we prioritize selecting nodes that have higher energy and larger degree to form the backbone. By transmitting data through the backbone with small routing cost, each node can preserve its energy effectively. Moreover, the nodes in the virtual backbone would not die quickly because of their high energy level. The proposed algorithm creates a small-size connected dominating set, which performs well in energy saving and efficiently affords more numbers of rounds in data collection.

The main contributions of this paper are summarized as follows.

First, we introduce the concept of energy-balanced connected dominating set in Section 3. Using this scheme, DGA-EBCDS can well balance energy consumption among nodes and consequently prolong the lifetime of the network.

Second, we theoretically analyze DGA-EBCDS algorithm in Section 4. Theoretical analyses prove that the set of the dominators is a maximal independent set (MIS) when the first stage for DGA-EBCDS terminates and the message complexity of DGA-EBCDS is $O(n \log n)$.

Third, we support our algorithm analysis with extensive simulations in Section 5. Compared with another dominating set-based algorithm mr-CDS [17], the average energy consumption of DGA-EBCDS is reduced by 33.3% in the formation of CDS, and the network lifetime is prolonged by 57.8%.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the related work. Section 3 presents the system model and problem statement. In Section 4, we describe and analyze the proposed DGA-EBCDS protocol in detail. The simulation results and corresponding discussions are given in Section 5. Finally, Section 6 concludes the paper and presents the further research work.

2. Related Work

The construction of connected dominating set in wireless networks has been extensively studied for many years. In order to measure the quality of a connected dominating set, the size of the CDS is usually chosen as the main concern factor. A smaller CDS can suffer less interference from its neighbors because each node in a WSN shares the communication channel with its neighbors. Moreover, a smaller CDS can perform well in reducing the number and the cost of control messages in routing and transmission. On the other hand, the smaller CDS can also make the management of the virtual backbone easier. Based on the above reasons, many researches focused on reducing the size of a CDS [18–31]. Dai and Wu [18] firstly introduced two greedy algorithms to computer the MCDS in a general$G$; both of them were polynomial time algorithms. Some works [18–20, 22] focused on constructing k-connected m-dominating sets for fault tolerance, and most of these works used a UDG as their network model. Wu et al. [23] proposed a CDS construction algorithm using an energy model which was also adopted in other works [24, 25]. Cardei et al. [26] and Funke et al. [27] proposed another two distributed algorithms, which could achieve better performance. Authors in [28–30] presented a distributed algorithm for constructing CDSs in UDGs, which consisted of two phases. Thai et al. [31] addressed the problem of constructing a CDS in a heterogeneous network and presented two approximation methods to obtain MCDS. Reference [21] considered constructing CDS in heterogeneous network, which formed an energy-efficient virtual network by using directional antennas. These algorithms effectively reduced the backbone size and improved the performance of the network.

Besides these algorithms, there were many other CDS-based algorithms aiming at the optimization goal of reducing the energy consumption of nodes in order to make the network live longer. CDS-BD-D [32] considered how to construct diameter restricted minimum connected dominating set. The diameter referred to the maximum length of the shortest path between any two nodes in the CDS. Data collection could be finished in limited delays by constructing diameter-restricted MCDS. GOC-MCDS-D [33] considered the routing cost of nodes and achieved higher energy efficiency than CDS-BD-D. Wu_h [34] aimed at creating the minimum connected dominating set and ignored the diameter size of the CDS. Another classic protocol named mr-CDS [17] considered the residual energy of nodes and effectively improved the energy efficiency of Wu_h [34]. Firstly, the node with higher residual energy than its neighboring nodes had a higher priority to become the dominator node. The selected dominator broadcasted its dominant message to its neighbors. The dominatee nodes could infer the number of the dominators among their neighbors from these dominant messages. Then, each node that was not in the dominating set would broadcast its neighbors' dominating set, which enabled the 2-hop neighbors of the dominators to know the dominators. By comparing the set of the 2-hops neighbors, each dominatee node could judge if there were any paths for the 2-hop dominator nodes. If no path existed to connect the dominator nodes, the node itself would become a dominator node. Finally, all dominator nodes would form a connected dominating set. However, the process of a dominatee node changing to a dominator node could not consider the energy factor, and each dominatee node autonomously converted to a dominator node, which made too many dominatee nodes change to dominator nodes at the same time. This caused the nodes consuming too much energy to run the pruning algorithm so as to reduce the number of nodes in the CDS. The message complexity of mr-CDS is $O(n \log n)$. Since mr-CDS is very easy and effective, we will compare this classic protocol with our protocol in this paper.

Moreover, there were some other algorithms aiming at the optimization goal of ensuring the reliability of the network. That is to say, the data could still be transmitted to the sink after the death of some nodes. LDA [35] focused on how to structure$k$-connected m-dominating set, and each dominator node constructed a local k vertex-connected subgraph according to the neighbor information. Then, each dominator node noticed the dominate nodes in this subgraph to join the CDS. FT-CDS-CA [36] studied the problem of constructing fault-tolerant CDS and proposed a constant factor polynomial time approximation algorithm to compute (3, m)-CDS, which was similar to the method used in [22]. SSC [37] was the first self-stabilizing distributed (k, r) algorithm, which used synchronous multiple paths to improve security, availability, and fault tolerance of the algorithm.

3. System Model and Problem Statement

3.1. Network Model. Assume that there are n sensor nodes in the network that are labeled as $v_1, v_2, \ldots, v_n$, respectively. Denote the sink by v_0. All the nodes are randomly deployed in a $M \times M$ field to continuously monitor the environment. Denote the transmission range of sensor nodes as r. We assume that $r \ll M$. The sink and all the sensor nodes form a connected network. Let V be the set of nodes. $|V|$ represents the number of nodes in set V, so, $|V| = n + 1$. The network has the following characteristics:

(1) The network is static; that is, all nodes and the sink are stationary after deployment.

(2) Nodes may have different initial energy. The sink is assumed to have infinite power supply and powerful computation ability.

(3) Nodes are not aware of their geographic information.

(4) Data are highly correlated; the nodes in the virtual backbone (the nodes in the CDS) can aggregate the data into one packet of k bits for transmission no matter how much data are arriving. The data aggregation makes the node able to merge its own data with the received data from its neighbors, leading to significant energy and bandwidth savings.

In this paper, we consider a wireless sensor network where data is periodically reported from the sensor nodes to the sink. Data collection and transmission proceeds in rounds, the data from the sensor nodes are collected and aggregated with the packets of peer sensor nodes, and only one data is sent per round from the sensor network to the sink, so, each node has only one packet of information to communicate in each round. A round is defined as the process of collecting all the data from nodes to the sink, regardless of how much time it takes [38].

3.2. Energy Model. DGA-EBCDS uses the same energy consumption model which is widely adopted in previous works [39, 40]. According to this model, the energy dissipated to deliver a packet of k bits from the source to the destination is defined as

$$E_t(k, r) = kE_{\text{elec}} + k\varepsilon_{\text{fs}} r^2, \quad (1)$$

where E_{elec} is the energy dissipated in operating the transmitter radio, r is the transmission range of the node and $\varepsilon_{\text{fs}} r^2$ represents the energy dissipated by transmitter amplifier which varies with the distance r between the two nodes. The energy dissipated in operating the receiver radio is expressed as

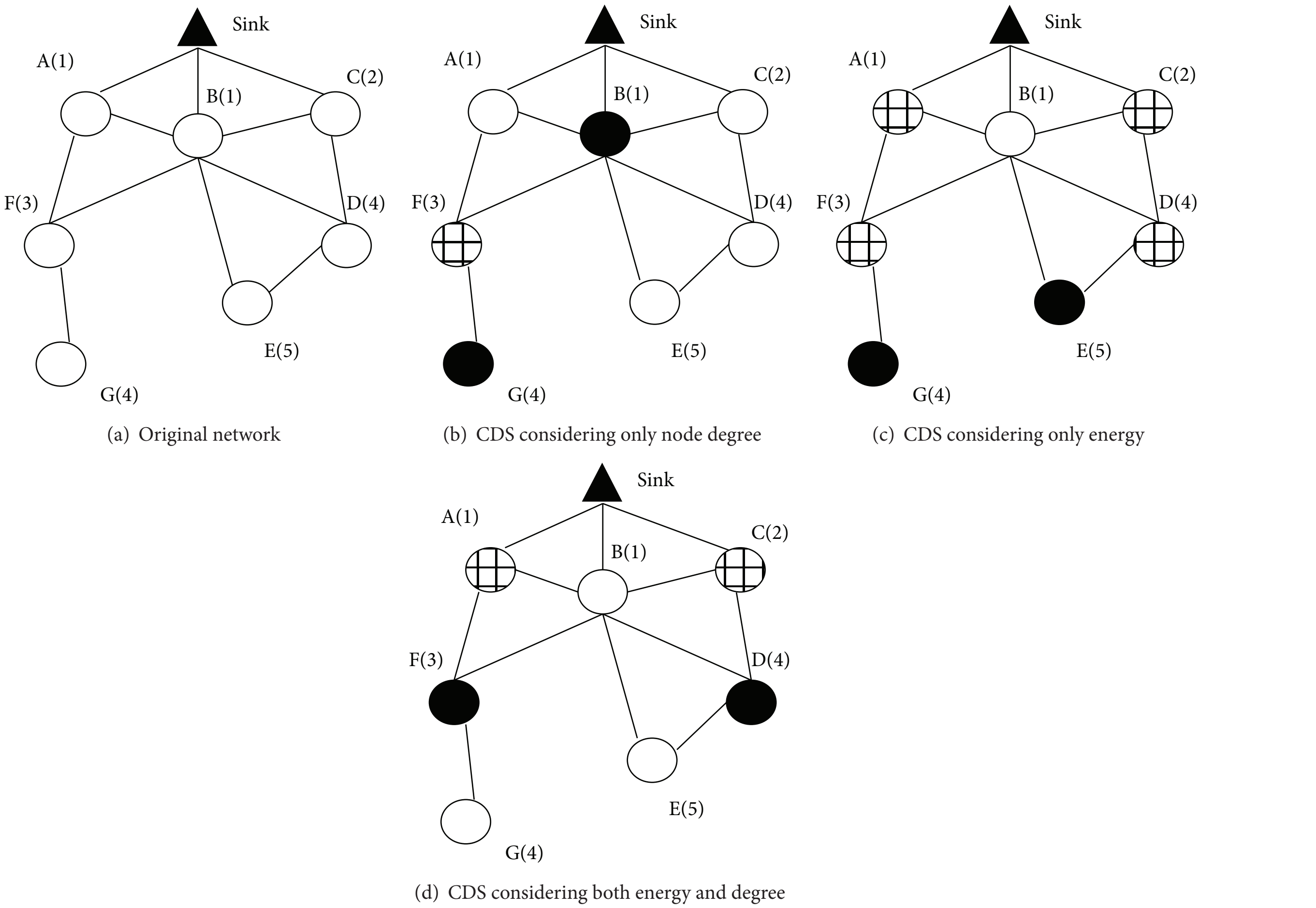
$$E_r(k) = kE_{\text{elec}}. \quad (2)$$

3.3. Problem Statement. The minimum connected dominating set (MCDS) is an NP-hard problem for general graphs [41]. The MCDS can simplify network abstracted topology and reduce routing cost. However, when the nodes in the MCDS are without enough energy, these nodes will die earlier because of consuming energy too quickly.

Consider the graph shown in Figure 1(a). There are 7 nodes and one sink in the network, and each node starts with different amount of energy. A (1) represents node A having one unit of energy, and so forth.

In Figure 1(b), network generates CDS only by node degree. B has the largest node degree, so it becomes a dominator (to be blacked) firstly. And the neighboring nodes of B (A, C, D, E, and F) become the dominatees. The left node G has no other nodes to compare with, so it becomes the dominator naturally. In order to connect two dominators G and B, F becomes the connector (to be shaded). So, CDS contains three nodes (B, F, and G), accounting for 43% of the total number of nodes, and there is only one path from the CDS to the sink; this path transmits all the data flows of the network; thus, the energy consumption of this path is very large.

In Figure 1(c), network generates CDS by node energy. E and G have larger energy, so they become dominators firstly, and then, A and F become the connectors to connect G and the sink, and C and D become the connectors to connect E and the sink. Thus, CDS contains six nodes, accounting for 86% of the total number of nodes. CDS as the virtual backbone has to work long hours and should not go to the sleep mode, so this part of the nodes will run out of energy quickly and die earlier.



(a) Original network

(b) CDS considering only node degree

(c) CDS considering only energy

(d) CDS considering both energy and degree

Figure 1: Different CDS-generating approaches (blacked nodes represent dominators, shaded nodes represent connectors).

In Figure 1(d), network generates CDS by two factors: both the energy and the degree of a node. F and D both have larger degree and higher energy, so they become dominators firstly, and then, A becomes the connector to connect F and the sink, and C becomes the connector to connect D and the sink. Thus, CDS contains four nodes, accounting for 57% of the total number of nodes. Meanwhile, there are two paths from the CDS to the sink; more nodes outside the CDS can go to the sleep mode, which can not only reduce the size of the CDS, but also balance the energy consumption among nodes and extend the network lifetime.

So, constructing an energy-balanced CDS is a good method to solve the problems of MCDS, which takes both the energy and the degree of a node into consideration so as to effectively prolong the network lifetime in wireless sensor network.

4. The Design and Analysis of DGA-EBCDS

For the sake of brevity in describing our algorithm in the following, we give some definitions and notations here. Let $N(v_i)$ be the neighbors set of a node i. Dominators represent the nodes in the dominating set. Dominatees represent the nodes that are not in the dominating set. Connectors refer to the nodes that can connect two different dominators that are not adjacent to each other, and all the dominators and connectors can form a connected dominating set. Independent set (IS) is a subset of V such that no two vertices within the set are adjacent in V. Maximal independent set (MIS) is an independent set such that adding any vertex not in the set breaks the independence property of the set; thus, the number of the IS nodes in the MIS is the largest. Let $P(v_i)$ be the set of neighboring nodes of node i that cannot become any kinds of dominators, dominates, or connectors. These definitions and notations will be used throughout this paper.

DGA-EBCDS is a distributed algorithm it depends only on the local information of nodes. The algorithm does not require the global information. DGA-EBCDS includes two stages: in the first stage, DGA-EBCDS selects the dominators; in the second stage, DGA-EBCDS selects the connectors, and finally, all the dominators and connectors form a CDS.

In order to construct an energy-balanced CDS, we use a weight W_i to measure the capability of a node, $W_i = E_i\text{Degree}_i$, where E_i is the energy of node v_i and Degree_i is the degree of node v_i. The higher the weight is, the more is its chance to become a dominator. By combining energy and degree together to measure the capability of a node, the node with a higher weight has higher priority to become a

```
At the beginning of the first stage:
$P(v_i) = N(v_i)$;
 if ($W_i > \max\{W_j \mid v_j \in P(v_j)\}$) || (($W_i == \max\{W_j \mid v_j \in P(v_j)\}$)&&($v_i == \min\{v_i, \{v_k \mid W_k = \max\{W_j \mid v_j \in P(v_j)\}\}\}$)))
 {  $v_{i.}$ color = "Black";
     $v_i$ broadcasts a message dominator($v_i$)
When $v_i$ receives a dominator ($v_k$) message:
if ($v_{i.}$ color == "White")
  {$v_{i.}$ color = "Gray";
      $v_i$ broadcasts a message dominate($v_i$);}
When $v_i$ receives a dominate ($v_j$) message:
if ($v_{i.}$ color == "White")
{  delete $v_j$ from $P(v_i)$;
 if ($W_i > \max\{W_j \mid v_j \in P(v_j)\}$) || (($W_i == \max\{W_j \mid v_j \in P(v_j)\}$)&&($v_i == \min\{v_i, \{v_k \mid W_k = \max\{W_j \mid v_j \in P(v_j)\}\}\}$)))
   {  $v_{i.}$ color = "Black";
            $v_i$ broadcasts a message dominator($v_i$);  }}
```

Algorithm 1: The first stage for DGA-EBCDS.

backbone node. That is to say, the node with more energy and larger degree has more chance to become a CDS node. In this section, we will describe the details of DGA-EBCDS as follows.

4.1. Construction of Energy-Balanced CDS. Sensor nodes in the DGA-EBCDS can have three different colors: "white," "gray," and "black". Initially, all nodes are "white." During the execution of our algorithm, all nodes change their color to either "black" or "gray." Based on the weight comparison among neighbors, some suitable nodes are selected as dominators; nodes that are not in the dominating set remain as dominatees. All the black nodes form the dominating set (network backbone), whereas the gray nodes remain as dominatees, and they use neighboring dominators as next hops to transmit the data. Each node runs DGA-EBCDS distributedly and the algorithm includes the two following stages:

(1) Weight comparison among neighbors in the first stage, and some suitable nodes are selected as the dominators. During weight comparison, node i is more suitable to be a dominator than its neighboring j when node i has a higher weight than j, or the weights of node i and j are the same, but the ID of node i is smaller than that of node j. When a node i wins the weight comparison over its neighbors, the node i becomes the dominator and turns "black;" when its neighboring nodes receive the black message from node i, they will become dominatees and turn "gray." Algorithm 1 gives detailed description of the selection of dominators by comparing the weights among nodes.

(2) In the second stage, the dominators select the suitable neighbors to become the connectors, and then the dominating nodes can be connected by the connectors. The selection is decided on the energy level of their neighbors as well as the connection condition between the neighbors and other dominators. When node i becomes a connector, it will turn "black" too. Finally, all the dominators and connectors will form a connected dominating set (network backbone). We use two lists (List1 and List2) to save the path information. Algorithm 2 gives detailed description of the selection of connectors.

In Algorithm 2, at the beginning of the second stage, if the color of node v_i is "Gray," v_i will broadcast a message dominatee1(v_i, v_k) for each dominator v_k that is one of the neighbors of v_i; this message will help v_i to search for other dominators that v_i can connect them through v_k.

When v_i receives a dominatee1(v_i, v_k) message, v_i checks the color of itself; if the color of node v_i is "Gray," and if there is a dominator v_m that is one of the neighbors of v_i and $v_m < v_k$, v_i will broadcast a message dominatee2(v_i, v_j, v_k); the message will make v_m know that it can connect v_k through v_i and v_j. If the color of node v_i is "Black," v_i will wait to receive the dominatee1 and the dominatee2 messages being sent from other neighbors; otherwise, if there is a record $[v_l, v_k]$ in List1 and $E_l < E_j$, delete $[v_l, v_k]$ from List1 and then add $[v_j, v_k]$ to List1.

When v_i receives a dominatee2(v_n, v_j, v_k) message, if the color of node v_i is "Black," and if List2 is empty or $[v_m, v_l, v_k]$ cannot be found in List2, add the record $[v_n, v_j, v_k]$ to List2; v_i waits to receive the dominatee1 and the dominatee2 messages being sent from other neighbors; otherwise, if there is a record $[v_m, v_l, v_k]$ in List2 and $\min\{E_m, E_l\} < \min\{E_n, E_j\}$, delete $[v_m, v_l, v_k]$ from List2; and then, add the record $[v_n, v_j, v_k]$ to List2.

When the timer expires, if v_i can connect with v_k through v_1, add v_l to a set A and delete the record $[v_n, v_j, v_k]$ from List2; if $v_i \in A$ and the color of node v_i is "Gray," v_i will change itself from a dominatee to a connector, and then, the color of v_i turns to "Black."

Each node runs DGA-EBCDS distributedly; when a node turns to "Black," it will never change its color again. At the end of the second stage, the set of "Black" nodes (dominators and connectors) forms a CDS.

```
At the beginning of the second stage:
if($v_{i.}$ color == "Gray")
   for (each dominator $v_k$ that is one of the neighbors of $v_i$)
      $v_i$ broadcasts a message dominatee1($v_i$, $v_k$);
When $v_i$ receives a dominatee1($v_j$, $v_k$) message:
if ($v_{i.}$ color == "Gray")
  {  if (there is a dominator $v_m$ that is one of the neighbors of $v_i$)&&($v_m < v_k$)
     $v_i$ broadcasts a message dominatee2($v_i$, $v_j$, $v_k$);
       }else if ($v_{i.}$ color == "Black")
           {  if ((List1 == null)‖([$v_l$, $v_k$] is not in List1))
  {   add record [$v_i$, $v_k$] to List1;
       $v_i$ waits to receive the dominatee1 and the dominatee2 messages sending from other neighbors;
  }else if (there is a record [$v_l$, $v_k$] in List1 and $E_l < E_j$)
    {   delete [$v_l$, $v_k$] from List1;
          add [$v_j$, $v_k$] to List1;}}
When $v_i$ receives a dominatee2($v_n$, $v_j$, $v_k$) message:
if ($v_{i.}$ color == "Black")
  {  if ((List2 == null)‖([$v_m$, $v_l$, $v_k$] is not in List2))
      {  add record [$v_n$, $v_j$, $v_k$] to List2;
        $v_i$ waits to receive the dominatee1 and the dominatee2 messages sending from other neighbors;
      }else if (there is a record [$v_m$, $v_l$, $v_k$] in List2)&&(min {$E_m$, $E_l$} < min {$E_n$, $E_j$})
        {  delete [$v_m$, $v_l$, $v_k$] from List2;
               add [$v_n$, $v_j$, $v_k$] to List2;   }}
When the timer Timer expired:
for (each record [$v_l$, $v_k$] in List1)
{  add $v_l$ to a set $A$;
   delete [$v_n$, $v_j$, $v_k$] from List2;  }
for(each record [$v_n$, $v_j$, $v_k$] in List2)
{    add $v_n$ and $v_j$ to the set $A$;          }
$v_i$ broadcasts a message connector ($A$, 2);
When $v_i$ receives a connector ($A$, $a$) message:
$a = a$-1;
If ($v_i \in A$)&&($v_{i.}$ color == "Gray")
   $v_{i.}$ color = "Black";
if ($a \geq 1$)
   $v_i$ broadcasts a message connector ($A$, $a$);
```

Algorithm 2: The second stage for DGA-EBCDS.

4.2. Analysis of the Distributed Algorithm

Theorem 1. *The set of the dominators is a maximal independent set (MIS) when the first stage for DGA-EBCDS terminates.*

Proof. From the detailed description of the algorithm shown in Algorithm 1, all the nodes will make a weight comparison among neighbors in the first stage of DGA-EBCDS. During weight comparison, v_i will be selected as a dominator when v_i has a higher weight than its neighboring v_j, or the weights of v_i and v_j are the same, but the ID of v_i is smaller than that the ID of v_i. When v_i wins the weight comparison over its neighbors, it will become a dominator.

When v_i becomes a dominator, all the neighbors of v_i will become dominates; thus, each two dominators cannot be connected directly. From the above analysis, the set of all the dominators is an independent set (IS). For any node $v_i \in V$, according to DGA-EBCDS, there are only two possible cases for v_i: if v_i is not a dominator, it must be a dominatee. For each dominatee, there is certainly at least one dominator in its neighbors. Therefore, any dominatee turning to a dominator will break the independence property of the current set (produce two dominators that can be connected directly). Thus, the set of the dominators is a maximal independent set (MIS) when the first stage for DGA-EBCDS terminates. □

Theorem 2. *The message complexity of DGA-EBCDS is* $O(n \log n)$.

Proof. In order to guarantee that the network is connected, the transmission radius of r should satisfy the following formula [42]:

$$r \geq \frac{\sqrt{2}}{2} M \sqrt{\frac{1}{n} \log\left(\frac{n}{\delta}\right)}, \tag{3}$$

where δ represents the probability that the network is not connected. Certainly, the number of the neighboring nodes must satisfy

$$N_{\text{neighborhood}} \approx \frac{n\pi r^2}{M^2} \geq \frac{\pi}{2} \log\left(\frac{n}{\delta}\right) = O\left(\log n\right). \quad (4)$$

From the detailed description of the algorithms shown in Algorithms 1 and 2, for each node v_i, if v_i is a dominator, it will send two messages: one message for declaring itself being selected as the dominator and the other message for broadcasting the set of the connectors.

If v_i is a dominatee, it will send at most $O(\log n)$ messages: one message for broadcasting dominatee(v_i), $O(\log n)$ messages for broadcasting dominatee1(v_i, v_j), and $O(\log n)$ messages for broadcasting dominatee2(v_i, v_j, v_k), where $O(\log n)$ is decided by the number of the neighboring nodes of v_i.

If v_i turns out to be a connector, it will broadcast connector (A, a) message only once.

Thus, a node will send at most $O(\log n)$ messages in DGA-EBCDS. There are n nodes in the network; we can deduce that the total message complexity of the network is $O(n \log n)$, which is the same as the message complexity of the classic DS-based algorithm mr-CDS. □

5. Performance Evaluation

We conduct extensive simulations to evaluate the performance of DGA-EBCDS. The experiments are assumed to be performed in a square field of $M \times M$, in which nodes are randomly dispersed. The parameters are as follows.

Each node in the field is assigned a randomly-generated initial energy level between 1 joule(J) and 1.5 J, $E_{\text{elec}} = 50$ nJ/bit, $\varepsilon_{\text{fs}} = 13$ pJ/bit/m^2, and each node will generate only one packet of 1000 bits per round. We compare DGA-EBCDS with mr-CDS and use the same assumption of mr-CDS. Sink is located at the center of the field; its coordinate is $(0.5M, 0.5M)$. All experiments will be performed 20 times, and their average values are taken as the final results. Our target is to balance the energy among nodes and delay the death of the first node. Network density is defined as the number of nodes lying in each unit of the field, which can be described as n/M^2. Average node degree represents the number of nodes in its transmission range, which can be described as $n\pi r^2/M^2$; we will use them to set our experiment scenes.

5.1. Backbone Size. We simulate a network of 480 static nodes placed randomly in a 1000 m × 1000 m area to observe the distribution of the CDS. The transmission range of nodes is set to be 100 m. The simulation results are shown in Figure 2.

In Figure 2, the sink is marked with a pentagram, the nodes in the CDS (including dominators and connectors) are marked with circles adding a plus sign, and dominatees are marked with circles. We can observe that the CDS produced by DGA-EBCDS and mr-CDS can both cover the whole network, which can effectively guarantee the network coverage.

In order to measure the validity of these two algorithms, we have used two different environments to randomly deploy the sensor nodes. Figures 2(a) and 2(b) are the first deployment environment, and Figures 2(c) and 2(d) are the second deployment environment.

We can see from Figures 2(a) and 2(b), in the first experiment, on average, 142 nodes constitute the mr-CDS backbone, whereas in case of DGA-EBCDS, there are only 79 nodes generated in the backbone; the backbone size of DGA-EBCDS is reduced by 44.4% on average.

Meanwhile, we can see from Figures 2(c) and 2(d), in the second experiment, on average, 146 nodes constitute the mr-CDS backbone, whereas in case of DGA-EBCDS, there are only 86 nodes generated in the backbone; the backbone size of DGA-EBCDS is reduced by 41.1% on average.

As backbone nodes must remain active during the network operating period, with fewer nodes in the backbone, DGA-EBCDS consumes less energy to keep the network in an operational state, and a smaller backbone in DGA-EBCDS can make more nodes (dominatees) go to a periodic sleep mode and effectively save the energy of the network. Moreover, no matter how the randomized sensor placements change, DGA-EBCDS always achieves better performance than mr-CDS.

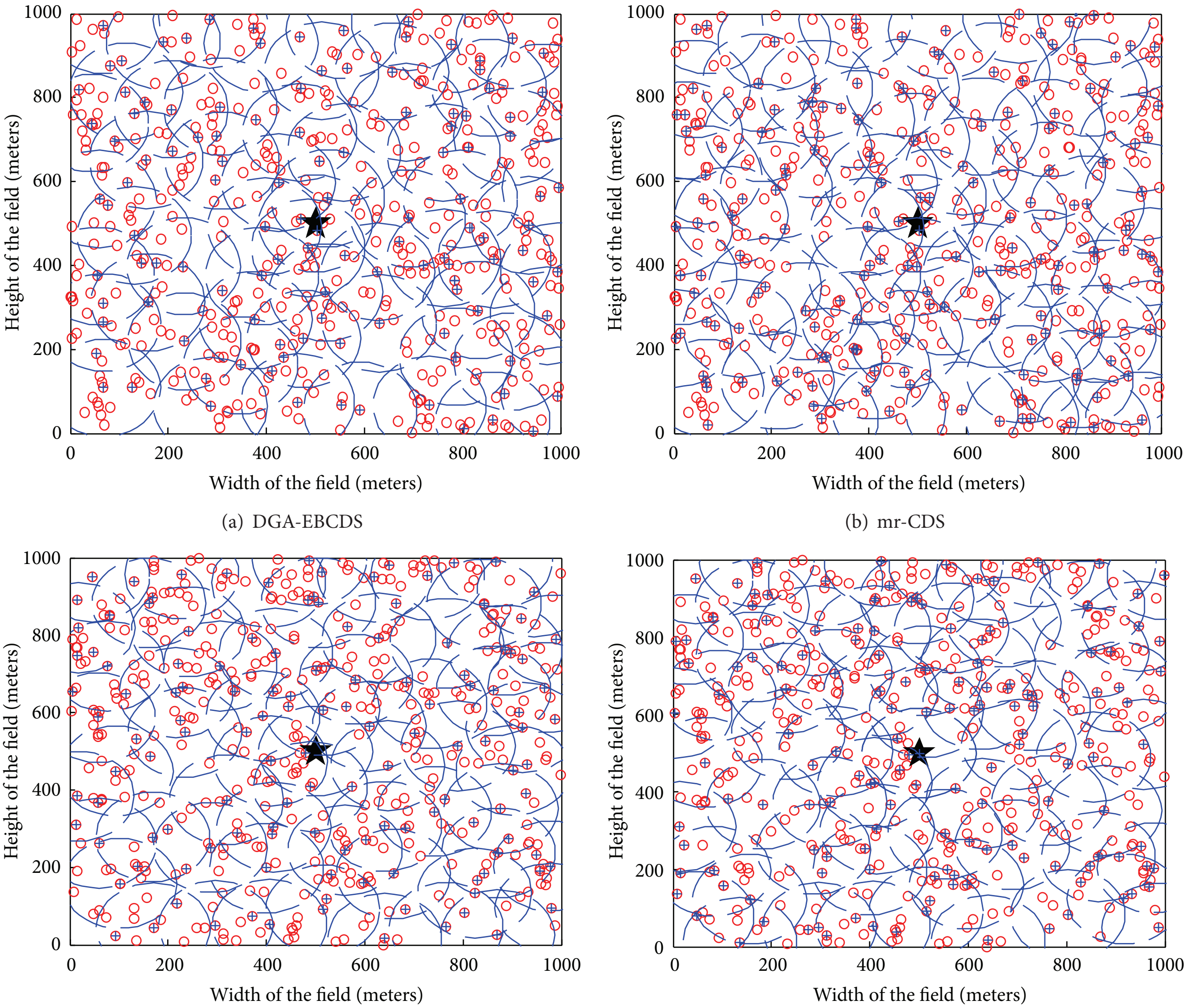
5.2. Energy Consumption in the Formation of CDS. In order to compare the energy efficiency of DGA-EBCDS and mr-CDS, we have examined the average energy consumption per node in the formation phase of CDS. The transmission range of nodes is set to 100 m, and the message packets size of nodes is 32 bits. We have investigated both algorithms under varying conditions like average node degree and the scale of the network. Two experimental scenes are used:

(1) Scene 1: the area is fixed to 1000 m × 1000 m. Assume that there are 320, 480, 640, 800, and 960 nodes randomly distributed in the field, respectively. The average node degree is 10, 15, 20, 25, and 30 under this scene, respectively.

(2) Scene 2: the average node degree is fixed to 15. The area is 800 m × 800 m, 1000 m × 1000 m, 1200 m × 1200 m, and 1500 m × 1500 m, respectively. There are 306, 480, 688, and 1075 nodes in this scene, respectively. The energy consumption in the formation phase of CDS is shown in Figure 3.

We can see from Figure 3(a), in the formation of CDS, the energy consumption of nodes increases when the average node degree grows. The reason is that nodes need to exchange messages with their neighbors. As the average node degree increases, the node needs to exchange its messages with more neighbors, which exhausts too much energy. However, DGA-EBCDS consumes less energy than mr-CDS; the average energy consumption has been reduced by 33.3%.

In Figure 3(b), when average node degree is fixed, the energy consumption for DGA-EBCDS and mr-CDS will be similar. Because the numbers of average neighbors of nodes will remain unchanged when average node degree is fixed,



(a) DGA-EBCDS

(b) mr-CDS

(c) DGA-EBCDS

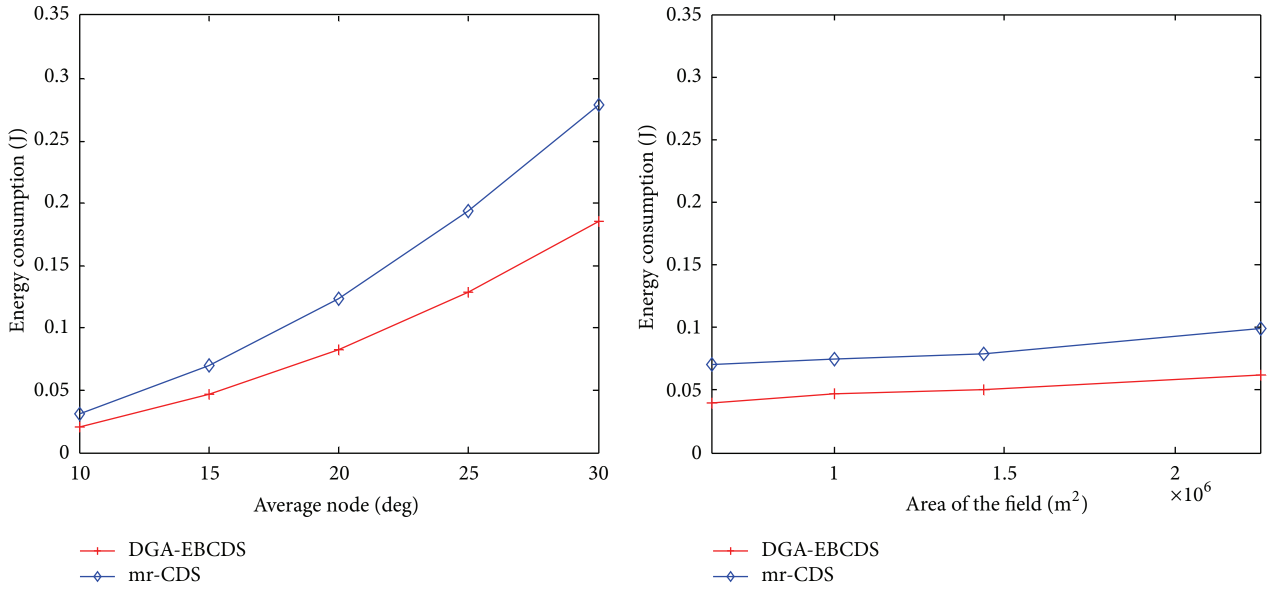
(d) mr-CDS

Figure 2: Backbone size comparison of DGA-EBCDS and mr-CDS.

the energy consumption will be unchanged. However, DGA-EBCDS consumes less energy than mr-CDS in different scales of network. Based on the observation, we can get the conclusion that DGA-EBCDS considers the energy balancing among nodes, as a result, DGA-EBCDS wastes less energy and acquires better energy efficiency than mr-CDS.

5.3. Network Lifetime. We set the same scenes of section B to evaluate the network lifetime. The energy of node will be reducing in data collection and the energy is changing per round. Node broadcasts its energy message at the beginning of each round. The message of energy can be delivered by some small packets; the energy consumption of this part can be neglected in the network.

In wireless sensor networks, the energy of node is very limited, if the network lifetime (FND) expires, remaining energy of the alive nodes cannot come to any use. So our algorithm tries to maximize the use of overall network energy before a network expires and extend network lifetime by delaying the death of the first node. In the process of data collection, the node cannot acquire the path information, so it does not know from which path it can transmit its data to sink with less energy consumption. There are many algorithms using the nodes in the CDS to construct a data collection tree and set the sink as the root of the tree. We no longer specially discuss this method here, and we use the distance vector (DV) routing algorithm [43] to construct a data collection tree after forming a CDS-based backbone. The collection tree is also used by mr-CDS. The sink acts as the root node and initiates tree formation, and only the dominators take part in the construction of the tree. Furthermore, we can see from Figure 2 that the number of dominators of DGA-EBCDS is quite lower when compared to the network size, so running



(a) Scene 1

(b) Scene 2

Figure 3: Energy consumption comparison of DGA-EBCDS and mr-CDS in the formation of CDS.

DV algorithm only over the dominators can reduce the cost of the network.

In the data collection process, each dominatee will transmit its data to the neighbor that has the highest energy in the data collection tree, and then it goes to the sleep mode to save its energy. The nodes in the data collection tree will aggregate the received data into one packet of 1000 bits and then transmit the packet to their father. A round of data collection terminates when the sink receives all the data from its children. Network lifetime achieved by these two algorithms is shown in Figure 4.

In Figure 4(a), the network lifetime is decreasing as the average node degree increases. DGA-EBCDS achieves longer network lifetime than mr-CDS no matter how the average node degree changes.

In Figure 4(b), when the average node degree is fixed, the network lifetime in large-scale network is shorter than that in small-scale network. As the transmission range is the same, the data need to be transmitted more times to get to the sink in large-scale network than in small-scale network, so the network lifetime in large-scale network will be reduced. However, DGA-EBCDS achieves longer network lifetime than mr-CDS. On average, the network lifetime is prolonged by 57.8%.

5.4. Delay. We set the same scenes of Section 5.2 to evaluate the data dissemination delays of these two algorithms. Since data dissemination delay are mainly affected by the maximum number of hops from the node to the sink, that is, the larger the number of hops, the longer the delay, we use the maximum number of hops between sink and the node to measure the data dissemination delay. The comparison results of data dissemination delay are shown in Figure 5.

In Figure 5, we can see that the data dissemination delays are mostly influenced by the size of the network; however, the average node degree produces very little influence on data dissemination delays.

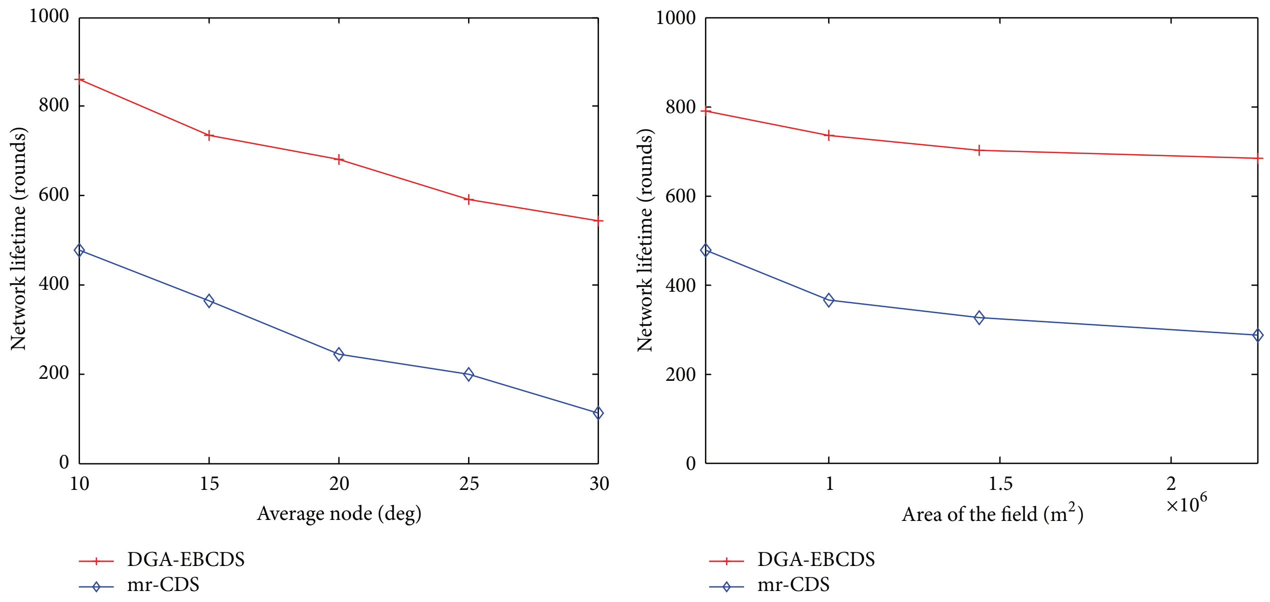
In Figure 5(a), the data dissemination delays of these two algorithms remain unchangeable as the average node degree increases. The average delay of DGA-EBCDS is 9.63 hops, and the average delay of mr-CDS is 8.36 hops. DGA-EBCDS incurs a larger delay (just only one hop on average) than that of mr-CDS.

In Figure 5(b), when the average node degree is fixed, the data dissemination delays of these two algorithms in large-scale network are larger than that in small-scale network. As the transmission range is the same, the data need to be relayed by more nodes to get to the sink in large-scale network than in small-scale network, so the data dissemination delays in large-scale network will be enlarged. The average delay of DGA-EBCDS is 10.49 hops, and the average delay of mr-CDS is 9.29 hops; mr-CDS shortens the delay by only one hop on average. Moreover, from Figure 2, we can see that the backbone size of DGA-EBCDS is smaller than that of mr-CDS. In some cases, there are fewer paths between the nodes and the sink, so some CDS nodes of DGA-EBCDS need to be transmitted more times to get to the sink; thus, DGA-EBCDS incurs larger delays than that of mr-CDS.

However, DGA-EBCDS considers the energy balancing among nodes. The algorithm can significantly delay the death of the first node, and the dominators can take more rounds for data collection. Based on the observation, we can draw the conclusion that DGA-EBCDS gains better performance than mr-CDS.

6. Conclusion and Future Work

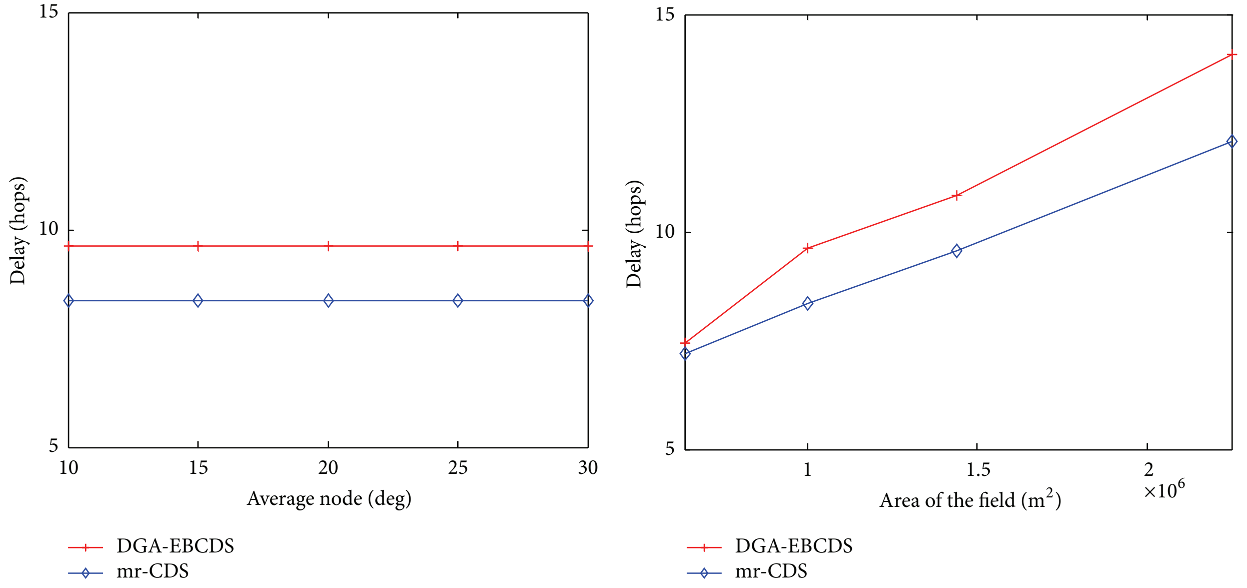
Selecting proper nodes to construct the CDS in order to maximize the network lifetime is an important issue when



(a) Scene 1

(b) Scene 2

Figure 4: Network lifetime comparison of DGA-EBCDS and mr-CDS.



(a) Scene 1

(b) Scene 2

Figure 5: Delay comparison of DGA-EBCDS and mr-CDS.

designing connected dominating set protocols and algorithms for data collection in wireless sensor networks. Most existing algorithms cannot consider the energy balancing among nodes, which limits their applications in large scale WSNs. The high contribution of this paper is the construction of an energy-balanced connected dominating set (DGA-EBCDS) for data collection in wireless sensor network. DGA-EBCDS can effectively increase the number of rounds before the first node failure by reducing the energy consumption per round and only choosing the nodes with a relatively higher weight to construct the CDS. Theoretical analyses show that the total message complexity of the algorithm is $O(n \log n)$, and the simulation results show that DGA-EBCDS can reduce energy consumption in the formation of CDS as well as effectively prolong the lifetime of the network.

In the next step, how to switch the role of dominators when designing the CDS backbone is a new research direction for us. If we always use the same set of dominators, those nodes must relay network traffic all the time, and they will easily run out of energy when compared with the dominatee nodes. We can consider that once the energy of a dominator goes below a threshold, a new node can take up the responsibility of this dominator. Therefore, researches that aim at delaying the death of the first node and switching the role of dominators can be useful in the future work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant nos. 61202494, 61103245), the Scientific Research Fund of Hunan Provincial Education Department (Grant no. 11C1099), the Natural Science Foundation of Guangxi (Grant no. 2012GXNSFBA053163), Hunan Provincial Science and Technology Department projects (2010FJ4052), and the Freedom Explore Program of Central South University (2012QNZT068).

References

[1] P. J. Chuang, Y. C. Yu, and C. S. Lin, "Reliable and congestion-controlling transport in wireless sensor networks," *Journal of the Chinese Institute of Engineers*, vol. 36, no. 1, pp. 2–16, 2013.

[2] A. F. Liu, Z. M. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multi-path routing for WSNs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3255–3265, 2012.

[3] N. Kimura, V. Jolly, and S. Latifi, "Energy restrained data dissemination in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2, no. 3, pp. 251–265, 2006.

[4] A. F. Liu and Z. H. Liu, "An elaborate chronological and spatial analysis of energy hole for wireless sensor networks," *Computer Standards and Interfaces*, vol. 35, no. 1, pp. 132–149, 2013.

[5] Y. Xu, W. C. Lee, J. Xu, and G. Mitchell, "Energy-aware and time-critical geo-routing in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 4, no. 4, pp. 315–346, 2008.

[6] K. Ota, M. Dong, Z. Cheng, J. Wang, X. Li, and X. Shen, "ORACLE: mobility control in wireless sensor and actor networks," *Elsevier Computer Communications*, vol. 35, no. 9, pp. 1029–1037, 2012.

[7] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen, "SMART: a secure multilayer credit-based incentive scheme for delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628–4639, 2009.

[8] H. Zhu, S. Du, Z. Gao, M. Dong, and Z. Cao, "A probabilistic misbehavior detection scheme towards efficient trust establishment in delay tolerant networks," *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[9] X. Y. Kui, J. X. Wang, and S. G. Zhang, "Energy-balanced clustering protocol for data gathering in wireless sensor networks with unbalanced traffic load," *Journal of Central South University*, vol. 19, no. 11, pp. 3180–3187, 2012.

[10] C. Zheng, S. X. Sun, and T. Y. Huang, "Constructing distributed connected dominating sets in wireless Ad Hoc and sensor networks," *Ruan Jian Xue Bao/Journal of Software*, vol. 22, no. 5, pp. 1053–1066, 2011.

[11] S. Gao and H. K. Zhang, "Optimal path selection for mobile sink in delay-guaranteed sensor networks," *Tien Tzu Hsueh Pao/Acta Electronica Sinica*, vol. 39, no. 4, pp. 742–747, 2011.

[12] S. Ji and Z. Cai, "Distributed data collection in large-scale asynchronous wireless sensor networks under the generalized physical interference model," *IEEE/ACM Transactions on Networking*, 2012.

[13] S. Ji and Z. Cai, "Distributed data collection and its capacity in asynchronous wireless sensor networks," in *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM '12)*, pp. 2113–2121, Orlando, Fla, USA, March 2012.

[14] A. F. Liu, P. H. Zhang, and Z. G. Chen, "Theoretical analysis of the lifetime and energy hole in cluster based Wireless Sensor Networks," *Journal of Parallel and Distributed Computing*, vol. 71, no. 10, pp. 1327–1355, 2011.

[15] 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, pp. 197–226, 2013.

[16] J. He, S. Ji, Y. Pan, and Z. Cai, "Approximation algorithms for load-balanced virtual backbone construction in wireless sensor networks," *Theoretical Computer Science*, 2012.

[17] S. Hussain, M. I. Shafique, and L. T. Yang, "Constructing a CDS-based network backbone for energy efficiency in industrial wireless sensor network," in *Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications (HPCC '10)*, pp. 322–328, Melbourne, Australia, September 2010.

[18] F. Dai and J. Wu, "On constructing k-connected k-dominating set in wireless networks," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS '05)*, Denver, Colo, USA, April 2005.

[19] F. Wang, M. T. Thai, and D. Z. Du, "On the construction of 2-connected virtual backbone in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1230–1237, 2009.

[20] Y. Wu, F. Wang, M. T. Thai, and Y. Li, "Constructing k-connected M-dominating sets in wireless sensor networks," in *Proceedings of the Military Communications Conference (MILCOM '07)*, pp. 1–7, October 2007.

[21] S. Yang, J. Wu, and F. Dai, "Efficient backbone construction methods in MANETs using directional antennas," in *Proceedings of the 27th International Conference on Distributed Computing Systems (ICDCS '07)*, pp. 45–50, June 2007.

[22] M. T. Thai, N. Zhang, R. Tiwari, and X. Xu, "On approximation algorithms of k-connected m-dominating sets in disk graphs," *Theoretical Computer Science*, vol. 385, no. 1–3, pp. 49–59, 2007.

[23] J. Wu, M. Gao, and I. Stojmenovic, "On calculating power-aware connected dominating sets for efficient routing in Ad Hoc wireless networks," in *Proceedings of the IEEE International Conference on Parallel Processing (ICPP '01)*, pp. 346–354, September 2001.

[24] J. Wu, B. Wu, and I. Stojmenovic, "Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets," *Wireless Communications and Mobile Computing*, vol. 3, no. 4, pp. 425–438, 2003.

[25] Y. Zeng, X. Jia, and Y. He, "Energy efficient distributed connected dominating sets construction in wireless sensor networks," in *Proceedings of the International Wireless Communications and Mobile Computing Conference (IWCMC '06)*, pp. 797–802, July 2006.

[26] M. Cardei, X. Cheng, X. Cheng, and D. Z. Du, "Connected domination in multihop Ad Hoc wireless networks," in *Proceedings of the 6th International Conference on Computer Science and Informatics (JC&I '02)*, pp. 251–255, March 2002.

[27] S. Funke, A. Kesselman, U. Meyer, and M. Segal, “A simple improved distributed algorithm for minimum CDS in unit disk graphs,” *ACM Transactions on Sensor Networks*, vol. 2, no. 3, pp. 444–453, 2006.

[28] P. J. Wan, K. M. Alzoubi, and O. Frieder, “Distributed construction of connected dominating set in wireless ad hoc networks,” in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’02)*, pp. 1597–1604, June 2002.

[29] K. M. Alzoubi, P. J. Wan, and O. Frieder, “New distributed algorithm for connected dominating set in wireless Ad Hoc networks,” in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS ’02)*, pp. 3849–3855, 2002.

[30] K. M. Alzoubi, W. Peng-Jun, and O. Frieder, “Distributed heuristics for connected dominating sets in wireless ad hoc networks,” *Journal of Communications and Networks*, vol. 4, no. 1, pp. 22–29, 2002.

[31] M. T. Thai, F. Wang, D. Liu, S. Zhu, and D. Z. Du, “Connected dominating sets in wireless networks with different transmission ranges,” *IEEE Transactions on Mobile Computing*, vol. 6, no. 7, pp. 721–730, 2007.

[32] D. Kim, Y. Wu, Y. Li, F. Zou, and D. Z. Du, “Constructing minimum connected dominating sets with bounded diameters in wireless networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 2, pp. 147–157, 2009.

[33] H. Du, Q. Ye, W. Wu et al., “Constant approximation for virtual backbone construction with Guaranteed Routing Cost in wireless sensor networks,” in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM ’11)*, pp. 1737–1744, Shanghai, China, April 2011.

[34] K. A. M. Almahorg, S. Naik, and X. Shen, “Efficient localized protocols to compute connected dominating sets for ad hoc networks,” in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM ’10)*, pp. 1–5, Miami, Fla, USA, December 2010.

[35] Y. Wu and Y. Li, “Construction algorithms for k-connected m-dominating sets in wireless sensor networks,” in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc ’08)*, pp. 83–90, HongKong, China, May 2008.

[36] D. Kim, W. Wang, X. Li, Z. Zhang, and W. Wu, “A new constant factor approximation for computing 3-connected m-dominating sets in homogeneous wireless networks,” in *Proceedings of the 29th IEEE Conference on Computer Communications (INFOCOM ’10)*, pp. 1–9, San Diego, Calif, USA, March 2010.

[37] A. Larsson and P. Tsigas, “A self-stabilizing, (k, r)-clustering algorithm with multiple paths for wireless Ad-hoc networks,” in *Proceedings of the 31st International Conference on Distributed Computing Systems (ICDCS ’11)*, pp. 353–362, Minneapolis, Minn, USA, June 2011.

[38] H. Ö. Tan and I. Körpeoğlu, “Power efficient data gathering and aggregation in wireless sensor networks,” *SIGMOD Record*, vol. 32, no. 4, pp. 66–71, 2003.

[39] X. Y. Kui, S. G. Zhang, J. X. Wang, and J. N. Cao, “An energy-balanced clustering protocol based on dominating set for data gathering in wireless sensor networks,” in *Proceedings of the 2012 IEEE International Conference on Communications (ICC ’12)*, pp. 193–197, Ottawa, Canada, June, 2012.

[40] 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, 2002.

[41] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, Calif, USA, 1978.

[42] V. Mhatre and C. Rosenberg, “Design guidelines for wireless sensor networks: communication, clustering and aggregation,” *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.

[43] R. Malhotra, *IP Routing*, O’Reilly, 1st edition, 2002.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 834245, 15 pages
http://dx.doi.org/10.1155/2013/834245

Research Article

An Energy-Efficient Cyclic Diversionary Routing Strategy against Global Eavesdroppers in Wireless Sensor Networks

Ju Ren, Yaoxue Zhang, and Kang Liu

School of Information Science and Engineering, Central South University, Changsha 410083, China

Correspondence should be addressed to Ju Ren; ren_ju@csu.edu.cn

Received 29 December 2012; Accepted 13 March 2013

Academic Editor: Mianxiong Dong

While many protocols for sensor network security provide confidentiality for message content, contextual information usually remains exposed, which can be critical to the mission of the sensor network. In this paper, we propose an energy-efficient cyclic diversionary routing (CDR) scheme against global eavesdroppers for preserving location privacy and maximizing lifetime of wireless sensor networks (WSNs). To ensure no impact on lifetime of WSNs, we minimize the energy consumption of hotspots and generate abundant diversionary routing paths in areas far from sink. To enhance the location privacy preservation, we theoretically figure out the probability of the cyclic interference routings in different areas of the network and statistically achieve an energy consumption balance in the entire network. Analysis and simulation results show that CDR significantly improves the security in source-location privacy preservation without reducing the network lifetime.

1. Introduction

A wireless sensor network consists of numerous cheap, small, and resource-constrained sensors that are self-organized to monitor and study the physical world. It bears a promising future in lots of important applications such as environment monitoring, military surveillance, and target tracking. Sensors collaborate to gather data and disseminate the data to sink [1, 2]. However, sensor networks are facing many security threats such as node compromising, routing disrupting, and false data injecting.

Among all these menaces, source-location privacy is of great interest since it cannot be addressed by classical security mechanisms, such as encryption and authentication. Take an example of event reporting in a sensor network. When a sensor detects an event, it transmits a message with event-related information to the sink. Simultaneously, the location of source sensor reveals itself to the adversary, who may be passively eavesdropping the traffic of the network, no matter how strong the data encryption key is [3–5]. The amount of communication overhead to be carried for preserving source-location privacy against eavesdroppers depends on what capabilities adversaries have. The possibility of the existence of a global eavesdropper who can monitor the entire network traffic leads to a great challenge for resource-constrained WSN. A high-assurance solution that can cope with such a powerful attacker adopts a periodic collection method where each sensor periodically transmits encrypted messages regardless of whether there is real data to send or not [5]. In general, periodic transmission together with encryption can effectively conceal the source of real packets from adversaries who do not have the decryption key. However, energy resource is relatively limited and redundant traffic contributes a great deal of energy consumption in WSNs. Thus, it is necessary to develop energy-efficient protocols to mitigate the overhead in periodic collection scheme. A feasible solution is that all sensors periodically generate packets but part of sensors played as proxies to filter dummy traffic.

Several similar methods have been proposed in the past, but the feasibility and the influence on network lifetime have yet to be explored. Since dummy traffic is supposed to be conducted in the entire network to confuse the global eavesdroppers, it definitely results in an explosion of the network traffic. If there are n sensors simulating the behavior of each source, the network lifetime will drop to $1/n$ of the original lifetime. Even with a dummy message filtering scheme, the traffic of the hotspots unavoidably expands to d

times, where d is the degree of the sink. In most of sensor networks, the value of d can usually run up to 5 or even more than 10. Such a formidable traffic expansion of the hotspots also means a sharp decrease in the network lifetime, let alone the traffic expansions in early schemes, in which every sensor of the network generates a fake message and sends it to the sink. Therefore, it is extremely necessary to design an efficient strategy combined with high security and maximal lifetime to improve the disappointing performance in network lifetime of the current techniques.

In this paper, we propose an efficient cyclic diversionary routing (CDR) scheme based on cluster structure to defend against global eavesdroppers in preserving location privacy and to maximize lifetime in wireless sensor networks. CDR is better than the existing studies in that CDR creates more fake sources to confuse adversaries than the traditional schemes, which greatly improves location privacy. At the same time, the network lifetime will not decrease even with an increase of diversionary routes compared with the traditional schemes. The major contributions of this paper can be summarized as follows.

(1) To the best of our knowledge, it is the first time that a source-location privacy preservation strategy has been proposed to defend against global eavesdroppers without sacrificing the lifetime of wireless sensor networks. Because network lifetime depends on the energy consumption of the hotspot and has no direct relationship to the whole network energy consumption, we can allow only real data flow to cross the hotspots and build cyclic diversionary routing paths in areas where the sensors have enough abundant energy to support them [6–9]. Furthermore, we analyze the energy consumption of different ring areas of the network based on our system model and figure out the probability of interference routing path in each ring. Therefore, the security of source-location privacy would be enhanced significantly, and meanwhile the network lifetime would be hardly affected despite the diversionary routing paths.

(2) In applications of the CDR protocol, we can achieve a significant enhancement in network security without reducing the network lifetime. At the same time, it also leads to some network latency in data transmission. Therefore, this protocol is especially suitable for applications that have urgent needs for network security while having moderate needs for network latency. However, in real-time network applications we can still provide trade-offs between location privacy security and data transmission latency.

(3) We conduct extensive simulation under OMNET++ of the proposed scheme CDR. Analysis and simulation results confirm the validity and efficiency of CDR and show that CDR not only significantly improves the security of source-location privacy but also achieves maximal network lifetime.

The rest of this paper is organized as follows. Section 2 reviews the related work. And the system model is described in Section 3. Section 4 presents the details of the CDR scheme. Security analysis and performance analysis are provided in Section 5. Section 6 shows the simulation results and their analysis and comparison. We conclude in Section 7.

2. Related Work

Privacy threats existing in sensor networks can be categorized into two types, namely, (i) content-oriented privacy threats and (ii) contextual privacy threats. The content-oriented privacy mainly concerns the ability of adversaries to crack the content of messages transmitted in sensor networks [1]. While encryption is an effective way to address these problems [10–12], adversaries are still able to extract the contextual information such as source-location from the messages. Therefore, contextual privacy should be studied more carefully due to their importance and urgency.

A number of techniques have been presented to dispose of contextual privacy threats. We can generally divide these works into two groups by the capability of the adversaries: strategies against local eavesdroppers and strategies against global eavesdroppers.

The local eavesdroppers have limited coverage, comparable to that of sensors. At any given time they can thus only monitor a local area. These attackers start from the sink and try to locate source node hop-by-hop in a tracing back way. Several methods have been proposed to address the source-location privacy problem against local eavesdroppers, for example, [4, 5, 13–15]. Kamat et al. proposed a classic and effective phantom routing scheme to solve this problem [13], which can be described as two phases. First, a message is sent through some neighbor sensors to a phantom node which is a random walk away. Then, the message will be either broadcasted or sent in the shortest path to the sink from the phantom node. Since each message probably traverses a different random walk path before being transmitted with greed routing, attackers are supposed to be confused by the diverse phantom source-location, and then the true source will be concealed.

However, the phantom nodes are usually extremely close to the real source, bringing a hidden danger in source-location preservation. Several advanced techniques have been proposed to improve the phantom routing strategy. In [16], researchers proposed a direct walk in section-based or hop-based approach to keep the phantom node away from the real source as far as possible. Li and Ren [4] developed three two-phase dynamic routing strategies to preserve source-location privacy. The main idea is to randomly transmit the message to a sensor far away from the actual source at first and then send the message to the sink with single path routing.

The strategies mentioned above all have outstanding performances in location privacy preservation against local eavesdroppers. However, global eavesdroppers are much more powerful and formidable adversaries compared with the local ones. For their ability of eavesdropping the entire network and traffic analysis, we have to devise more suitable and available schemes to defend source-location privacy against them. Part of related works is listed as follows.

Mehta et al. [17] first presented the global eavesdropper model. They formalized the source-location privacy issue under this strong adversary model and figured out the communication overhead needed for obtaining a given privacy. To address this problem, they proposed two strategies to preserve the location privacy: periodic collection and source simulation. The periodic collection method provides a high level of location privacy by making each sensor generate dummy traffic periodically. To reduce the enormous energy consumption caused by periodic collection, source simulation method randomly selects several sensors at multiple places to simulate the behavior of real objects to confuse the adversaries. However, fake sources also bring much extra energy consumption to the hotspots of the network, which leads to a poor network lifetime.

Shortly after that, several techniques have been presented to improve the network lifetime based on the idea of source simulation. Proxy filtering is one of the improved strategies. Yang et al. [18] illustrated the main idea of this scheme in a way that they select some sensors as proxies that proactively filter dummy messages on their way to the sink. Then, they proposed two methods named PFS and TFS to accurately locate proxies. Because the problem of optimal proxy placement is NP-hard, they employed local search heuristics with no guaranteed maximal network lifetime.

Recently, Bicakci et al. [19, 20] studied the lifetime in various proxy assignment schemes and different deployment scenarios. They propose a filtering method called OFS (Optimal Filtering Scheme) to maximize network lifetime and preserve source-location privacy against global eavesdroppers. This scheme is based on a Linear Programming framework. They claimed that Linear Programming is an effective method to find the optimal locations of proxies under a set of linear constraints.

Most of the techniques mentioned above can well preserve the location privacy against global eavesdroppers, but fixing the optimal proxy locations is not an easy task, and the network lifetime will be reduced more or less for the intrinsic disadvantage of the locating method. To address this problem, our CDR scheme adopts cluster structures to construct cyclic interference routing paths, in which the cluster heads will act as proxies to filter the fake messages generated by the fake sources. Additionally, we also theoretically figure out the probability of the cyclic interference routing paths in different areas of the network to provide optimal privacy protection without deteriorating the network lifetime.

3. System Model and Problem Statement

3.1. Network Model. We make the following assumptions about our network model.

(1) The wireless sensor network consists of sensor nodes that are uniformly and randomly deployed in a sensor field with density ρ, and they cannot move after being deployed. The sink is located at the center of the network and works as the network controller to collect event data.

(2) We assume the object is equipped with a GPS so that the sink can always be aware of its location. And the appearance of the object is randomly distributed in the entire network, so the probability that each sensor detects the information of the object is equivalent. We also assume that adversaries cannot attack the object within the area that is one hop away around the sink for the powerful monitoring ability in this area.

(3) The sensor nodes are assumed to know their relative locations and the sink location. That is, each sensor node has the knowledge of its neighbor nodes [21]. We also assume that a security infrastructure, such as powerful encryption, has already been built in; that is, no information carried in the message (e.g., packet head) will be disclosed. The key management, including key generation, key distribution, and key update, is beyond the scope of this paper.

3.2. Adversary Model. The adversaries are assumed to be external, passive, and global attackers. By external we mean that the adversaries will not compromise or control any sensors; by passive we assume that the attackers do not conduct any active attacks such as traffic injection, channel jamming, or denial of service attack; by global we presume that the adversaries can collect and analyze all the communications in the network. Note that it does not necessarily mean such global attackers are capable of detecting all the occurrence of real events in any place of the network by themselves, because (i) real event detection devices are often costly, whereas message collection devices are inexpensive and off the shelf; (ii) real event detection devices such as animal-monitoring cameras normally do not have sizes as small as regular sensors which means they are easy to be found and destroyed.

To be more specific, the adversaries may launch the following attacks in our model. On the one hand, even with encryptions of the messages in the network, it is still easy for the adversaries to trace back to the previous source of the messages if the encrypted messages remain the same during their forwarding process. On the other hand, the adversaries may perform more advanced traffic analysis including rate monitoring and time correlation. In a rate monitoring attack, the adversaries pay more attention to the nodes with different (especially higher) transmission rates. In a time correlation attack, the adversaries may observe the correlation in transmission time between a node and its neighbor, attempting to deduce a forwarding path.

3.3. Energy Consumption Model. Sensors consume energy when they are sensing the environment and receiving or transmitting data. The amount of energy consumed for sensing is not related to routing. Therefore, we consider only the energy consumption in transmitting and receiving messages. According to the radio model used in [6], energy consumption for transmitting is given by

$$E_t^{l,d} = \begin{cases} lE_{\text{elec}} + l\varepsilon_{fs}d^2 & \text{if } d < d_0, \\ lE_{\text{elec}} + l\varepsilon_{\text{amp}}d^4 & \text{if } d > d_0, \end{cases} \quad (1)$$

where E_{elec} is the transmitting circuit loss. Both the free space (d^2 power loss) and the multipath fading (d^4 power loss) channel models are considered in the model, depending on the distance between transmitter and receiver. ε_{fs} and ε_{amp} are the energy required by power amplification in these two models, respectively. The energy spent in receiving a l-bit packet is

$$E_r^l = lE_{\text{elec}}. \tag{2}$$

The above parameter settings are given in Table 1 [5].

For a better understanding of this paper, we detail the meanings of related notations in Table 2.

3.4. Problem Statement. It is a very challenging task to provide source-location privacy under the global adversary model. To prevent the traffic analysis attacks, trade-offs between various performance and security metrics such as privacy, latency, and network lifetime widely exist. If all the packets in the network are real event packets and every node reports, receives, or forwards a real event message immediately, it would be quite easy for a global attacker to trace back to the real source without any delay. Therefore, diversionary routing paths are necessary in the sensor network. But apparently, diversionary routing paths will significantly increase the network traffic, which is undesirable for sensor networks where communication overhead dominates the entire energy consumption. To guarantee the source-location privacy without reducing the network lifetime, we conclude our goals into the following two aspects.

(1) The proposed scheme should be secure enough to defend location privacy against the global adversaries; that is, the adversaries should not be able to get the source-location information by analyzing the traffic pattern in any phase of a data gathering period.

(2) The network lifetime should be scarcely affected by the diversionary routing paths. Since the sensors in the network are hard to recharge after deployment, the maximal network lifetime is the foremost goal in most applications. And for the reason that it is hard to minimize the event reporting delay along with diversionary routing paths, the proposed scheme should be best suitable for applications where a certain amount of transmission delay could be tolerated.

4. The Cyclic Diversionary Routing Scheme

In this section, we describe our *Cyclic Diversionary Routing* (*CDR*) scheme for location privacy preservation and lifetime maximization in wireless sensor networks. The principles of CDR can be expressed as the following three points. (1) All the cyclic interference routing paths and real routing path are homogeneous that the adversaries cannot distinguish them by their shape or size. And we ensure that a number of sensors simulating the behavior of real sources to confuse the adversaries always exist in the network. Therefore, the security of the network is enhanced in any phase of a gathering period. (2) As network lifetime depends on the energy depletion of the hotspots, the proposed strategy will not lay any kinds of additional burden on the hotspots of the network. And the energy consumption in other areas increased by the diversionary routing paths should also be designed to be no greater than the energy consumption of the hotspots. (3) Making full use of the abundant energy in areas far from sink to generate interference routing paths as many as possible, because the sensors in these areas always remain much energy when the network dies. Based on these three principles, we can then get the maximal network lifetime and improve the security of network in one strategy.

4.1. Overview of the Proposed Scheme. To conduct dummy traffic without reducing the network lifetime in hiding real events, we divide the network into several rings according to the hop counts from the sensors to the sink and establish cyclic diversionary route at different levels with a variant probability, just as shown in Figure 1. The main idea can be described as the following aspects. (i) We first divide the rings in areas other than the hotspot area into uniform clusters and name one of cluster heads in the outmost ring as the promoter. (ii) In each period, the ring where the object appears (called event-ring) must establish cyclic diversionary route, while the hotspot (i.e., the first ring) will never create interference route and will only relay the real messages to the sink. And other rings are scheduled to establish cyclic diversionary route with a certain probability p_i. If the sensors of a ring are notified to create interference routing, they are supposed to send dummy messages to their cluster heads with a probability q. When the cluster head receives all the data in the cluster, it will dump all the dummy data, and only the cluster head that is in a cluster where a real event occurs will keep the real data. (iii) Finally, the promoter will start the data transmission to the sink with one initial dummy data package. When the data package comes to a ring which is scheduled to establish cyclic diversionary route, it will take a round trip and gather data of all the cluster heads in the ring. After the intercluster communication in this ring, it will be forwarded to the cluster head of the next inner ring. Otherwise, it will be forwarded directly to the cluster head of the inner ring. In this method, data package will be safely forwarded to the sink ring by ring. Pay attention that when the data package comes across with the cluster head where a real object exists, dummy data will be replaced by real data in the data package. Therefore, if a real event occurs, only real data can finally make to the sink.

Specifically, the advantages of CDR mainly lie in the following two aspects.

(1) Improvement on security: in order to effectively defend against global eavesdroppers, at any given time dummy events that are homogeneous with real events must exist in the network. And also, the clusters where cluster members send dummy data to their cluster heads are completely the same as the cluster with the real source. In this case, adversaries cannot distinguish the real source from ways of intracluster communication. After intracluster data aggregation, each cyclic diversionary route is generated in the same way. Therefore, adversaries still cannot track back to a specific cyclic diversionary route since intercluster communications are all the same. In general, we provide secure preservation of location privacy at any phase of our strategy.

(2) Enhancement on network lifetime: it is well known that there are hotspots near the sink in WSNs, and usually, after the first node dies, the network can no longer perform

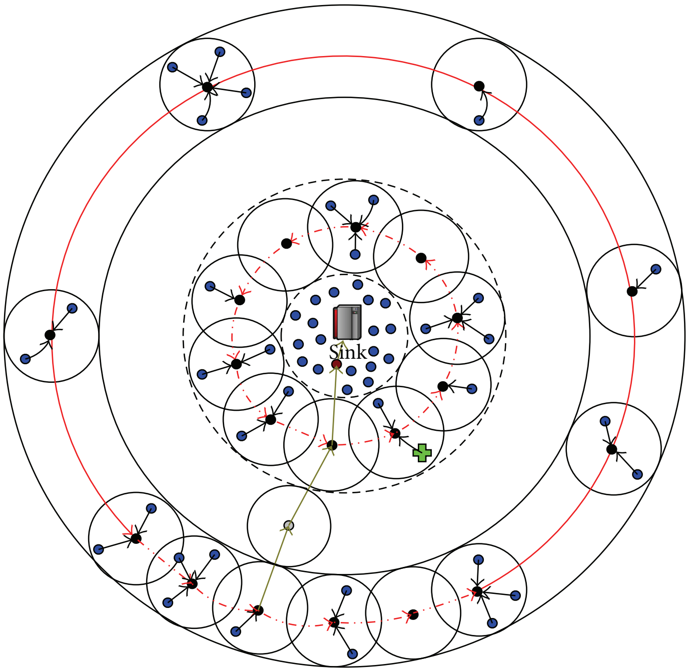


Figure 1: Illustration of CDR.

Table 1: Network parameters.

Parameter	Value
Threshold distance (d_0) (m)	87
E_{elec} (nJ/bit)	50
e_{fs} (pJ/bit/m^2)	10
e_{amp} (pJ/bit/m^4)	0.0013
Initial energy (J)	0.5

Table 2: Related notation meanings.

Notation	Meaning
R	Network radius (m)
r_t	Transmission radius (m)
n	The number of sensors
ρ	The node density of the network (sensors/m^2)
p_i	The probability of the ith ring is assigned to create cyclic diversionary route
q	The probability of a sensor is assigned to send dummy messages to the cluster head
σ	The number of data bits in a real message or in a dummy message

complete and effective monitoring of the entire network. Therefore, the network lifetime is usually defined as the lasting time from the beginning of the network till the time the first node dies [6]. Generally speaking, the security of location privacy is proportional to the amount of dummy traffic. While dummy traffic usually leads to a decrease in network lifetime. As we discussed in the section above, previous research mainly brings in a proxy filter scheme to provide trade-offs between network lifetime and security in which network lifetime can still be affected more or less. In CDR scheme, we adopt cluster heads to filter dummy packets in the cluster structure. In this case, despite the interference data generated in the whole network, intercluster communication flow will always keep at a moderate level. Moreover, we have no extra energy consumption of the hotspots and energy still remains in those rings even after generating the diversionary route. Theoretically, the CDR scheme can both preserve location privacy and maximize the network lifetime.

4.2. Description of Cyclic Diversionary Routing Scheme. To provide more detailed descriptions of our protocol, we divide the CDR scheme into three phases, (i) initialization and clustering; (ii) intracluster data aggregation; (iii) cyclic diversionary route establishment. In the following text, we will describe the procedures of the proposed cyclic diversionary routing scheme in detail.

(i) Initialization and Clustering. In our network model, we first divide the network into several rings according to the hop counts from the sensors to the sink. We assume that the sink has unlimited resources and works as the network controller to collect event data. Then we divide the rings in areas other than the hotspot area into uniform clusters. The clustering methods have been proposed in [22–25], and here we adopt the HEED clustering algorithm in [18], which can well balance the energy consumption of nodes in the same ring and the size of clusters in the network. Additionally, we select a cluster head in the outmost ring as the starting node of the cyclic diversionary route, which we call the promoter. And the token scheme presented in [26, 27] can be applied

here to randomly select the promoter in the outmost ring who holds the token at first and then passes it to the other cluster heads during clustering process.

At the initial stage of every period, rings other than the event-ring and the first ring are scheduled to establish cyclic diversionary route with a certain probability p_i. Based on this probability, the sink can learn about the routing information of different rings and broadcast this information to nodes in the network. And we call those rings assigned to create diversionary route interference rings.

(ii) Intracluster Data Aggregation. With the initial information, the cluster heads in interference rings will start gathering information among their cluster members. In this phase, the real source will send the real data to its cluster head, while other cluster members turn into fake sources with a certain probability p and then send dummy messages to their cluster heads.

Apparently, with more nodes involved in intracluster communication, we bring more confusion to the eavesdroppers, which results in a safer privacy of the network. However, from the energy perspective, more fake sources undoubtedly lead to more energy consumption of the network. In order to provide trade-offs between energy consumption and network security, here we randomly select part of the sensors to generate dummy messages and send the messages to their cluster heads, while cluster heads will then dump all the dummy messages they received and only keep the real message that was sent by a real source node. In this case, real event will be safely covered and real data will be successfully stored in the cluster heads of the interference ring.

(iii) Cyclic Diversionary Route Establishment. After the aggregation of intracluster data, the promoter in the outmost ring will start the data forwarding. At first, it will make a decision whether to generate interference routing or not based on the initial information. If an interference route is scheduled to create in this ring, the promoter will take its right neighbor as the next hop of data transmission, and the neighbor node will forward data from cluster head to cluster head around the whole ring in the same way. During the forwarding process, a cluster head that receives the data will check if it holds the real data or not. If so, it will drop the fake data and send the real data to the next hop, otherwise it will just relay the receiving data. Once the data has arrived at the promoter again, the cyclic diversionary route of this ring has finished. Then the promoter will find the cluster head which is the nearest to the sink in the adjacent inner ring and forward the data. This cluster head is supposed to make a judgement the same as the promoter or its previous node and take appropriate actions. While if no interference route is scheduled to generate in a ring, the promoter or the cluster head will just relay the data to the cluster head nearest to the sink in the adjacent inner ring directly. In this way, the data package will be forwarded to the sink orderly in the end. Figures 2 and 3 show the cyclic routing in a ring without or with a real event, respectively.

Through the above procedures, we can successfully generate interference routings in the whole network in one period to defend against global eavesdroppers. In the next period, we make minor adjustment in this scheme according to the location of the real event. As we know global adversaries have powerful abilities in rate monitoring and time correlation which means they can locate the event-ring by analyzing the frequencies of the interference routing generation in different rings. For example, if the adversaries notice that the xth ring has generated the interference routing in 10 periods in a row, they can basically deduce that the real object has stayed in this ring for a long time. Actually in practical applications, objects all have high probabilities to stop over at one place during a relatively long time. In this case, we introduce adjustment in our scheme to defend against the statistics attack of global eavesdroppers.

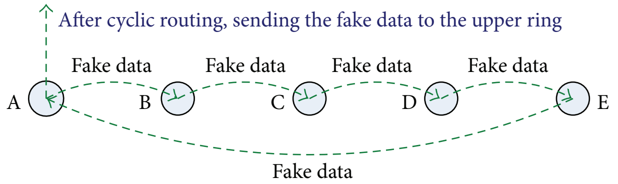

After cyclic routing, sending the fake data to the upper ring
Fake data
Fake data
Fake data
Fake data
A
B
C
D
E
Fake data

FIGURE 2: Cyclic route in the rings without real event.

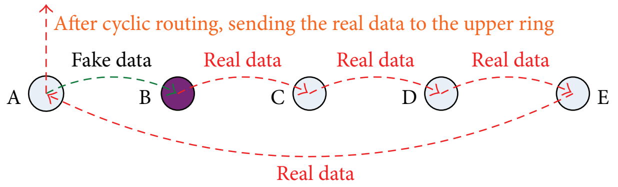

After cyclic routing, sending the real data to the upper ring
Fake data
Real data
Real data
Real data
A
B
C
D
E
Real data

FIGURE 3: Cyclic route in the ring with real event.

Broadcast scheme of the sink in CDR can well resist this kind of attacks. If the object stays in the same ring, we also generate interference routings in the same rings exactly as in last period. The adversaries can no longer perform statistics attack since the data flow in the whole network nearly remains the same. The Pseudocode of CDR algorithm is provided in Algorithm 1.

4.3. Energy Consumption Analysis. According to the above details of the CDR protocol, we see that CDR can effectively confuse the adversaries with the redundant traffic which may also lead to huge extra energy consumption on the whole network. We have to theoretically figure out the probability p_i of a ring to generate interference routing and the probability q of a cluster member to send dummy messages in this interference ring. With greater p_i and q, we will certainly have better network security along with much more dummy traffic. We have to further improve the network security in the precondition that extra energy consumption in the rings will not reduce the network lifetime.

Along with the three procedures of CDR establishment described above, the network energy consumption in CDR also mainly consists of three parts: clustering, intracluster data aggregation, and cyclic diversionary routing establishment. In one data aggregation period, cluster members send dummy messages with a certain probability and not too much data flow will send through the cluster head. In this

```
Cyclic Diversionary Routing
Phase I: Initialization (in sink)

Let $L_p$ denote the ring where the objective exists and Inter$f_{L_0} \leftarrow 0$, Inter$f_{L_p} \leftarrow 1$;
For each level $L_i$ of all the levels except $L_0$ and $L_p$ do
    Decide the value (0 or 1) of Inter$f_{L_i}$ with a probability of $p_i$;
End for
Randomly choose a node $N_k$ of the outmost ring to hold the token of promoter, for example $pt = N_k$;
Broadcast the Inter$f$ and $pt$ to all of sensors.

Phase II: Intra-cluster data gathering (in each sensor)

All of sensors except those in the hotspot begin to cluster with HEED clustering algorithm;
After clustering, the node with the token of promoter is supposed to deliver the token to his cluster head.
For each cluster member of each cluster do
    Decide to transmit dummy message to his cluster head with a probability of $q$;
End for.

Phase III: Cyclic diversionary route establishing

For each sensor $t$ that stays in each ring $s$ do
    trans = 0;
    If $t$ is the cluster head with the token of promoter then
        If Inter$f_{L_s}$ == 1 then
            $t$ transmits his data to his nearest neighbor cluster head which is in the same ring and at the right side of $t$.
        Else
            $t$ transmits his data to the cluster head which is the nearest to the sink among the cluster heads in the lower ring;
        End if
        trans = 1;
    Else
        If $t$ has received data from the upper ring then
            If Inter$f_{L_s}$ == 1 then
                If $t$ has the real data, then it drops the received data and transmits his real data to the next hop of the same ring;
                otherwise, it just relays this message to the next hop.
            Else
                If $t$ has the real data, then it drops the received data and transmit his real data to the lower ring; otherwise, it just
                relay this message to the lower ring;
            End if
            trans = 1;
        Else if $t$ has received data from the same ring then
            If trans == 1 then
                $t$ transmits his data to the lower ring;
            Else
                If $t$ has the real data, then it drops the received data and transmit his real data to the next hop of the same ring;
                otherwise, it just relay this message to the next hop.
            End if
            trans = 1;
        End if
    End if
End for
```

Algorithm 1: CDR protocol Pseudocode.

case we can extend the cluster head rotation period a little longer and it will not consume much energy in clustering in one data period. Moreover, according to our clustering algorithm, each node has the same probability to be a cluster head. Then statistically, all the nodes consume the same amount of energy in clustering after certain data periods. As a matter of fact, we can define the energy consumption in clustering as a small and fixed number in each period. Thus the energy consumption in intracluster data aggregation and cyclic diversionary routing establishment will be the primary factor affecting network lifetime. We will discuss the nodal energy consumption in the different rings of the network in the following theorems.

Theorem 1. *Assuming the nodal energy consumption in clustering is E_c, then, for the nodes in the ith ring of the network,* *the average nodal energy consumption in each period E_i^{avg} can be calculated as*

$$E_i^{\text{avg}} = \begin{cases} \dfrac{E_t^{\sigma,r/2} + E_r^{\sigma}}{\pi r^2 \rho}, & \textit{when } i = 1, \\ E_c + \dfrac{p_i q \left(N_i - N_i^c\right)\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right)}{N_i} \\ \quad + \dfrac{p_i N_i^c \left(E_t^{\sigma,r} + E_r^{\sigma}\right) + E_t^{\sigma,r} + E_r^{\sigma}}{N_i}, & \textit{when } i \neq 1, \end{cases} \tag{3}$$

in which N_i and N_i^c can be represented as the following equations:

$$\begin{gathered} N_i = S_i \rho = \pi\rho\,(2i-1)\,r^2, \\ N_i^c = \frac{S_i}{S_c} = \frac{\pi\,(2i-1)\,r^2}{\pi (r/2)^2} = (8i+4). \end{gathered} \tag{4}$$

Proof. At first, let us focus on the nodal energy consumption in the first ring where the nodes are nearest to the sink. In each data collection period, only one route is generated in this circular area. The average data transmission distance can be defined as $r/2$. Then the whole energy consumption in this area is $E_1^{\text{total}} = E_t^{\sigma,r/2} + E_r^{\sigma}$, and the average nodal energy consumption is $E_1^{\text{avg}} = E_1^{\text{total}}/\pi r^2 \rho = (E_t^{\sigma,r/2} + E_r^{\sigma})/\pi r^2 \rho$.

As for the energy consumption in other rings, we can figure it out in combination of two parts.

(1) Energy consumption in intracluster communication: the whole network has been divided into several circular rings with the same width r and the area of the ith ring is $S_i = \pi(ir)^2 - \pi((i-1)r)^2 = \pi(2i-1)r^2$, and the number of nodes in the ith ring is $N_i = S_i\rho = \pi\rho(2i-1)r^2$. According to clustering method we divide the rings into uniform clusters with a radius $r/2$; then the area of each cluster is $S_c = \pi(r/2)^2$ and the cluster number in each ring is $N_i^c = S_i/S_c = \pi(2i-1)r^2/\pi(r/2)^2 = (8i-4)$. If a ring has no schedule to generate interference route, then energy consumption in this part is zero. Otherwise, we can figure out the energy consumption in intracluster data aggregation in the following way. Each cluster member generates a dummy message with a probability q and sends the dummy message to the cluster head, and the data transmission distance is half of the cluster radius. Energy consumption in one cluster is $e_t^{\text{in}} = qE_t^{\sigma,r/2}$. And the energy consumed by the cluster head when receiving an intracluster data is $e_r^{\text{in}} = E_r^{\sigma}$. Hence, the nodal energy consumption in intracluster communication can be calculated as

$$\begin{aligned} e_i^{\text{in}} &= \frac{\left(N_i - N_i^c\right)\left(e_t^{\text{in}} + q e_r^{\text{in}}\right)}{N_i} \\ &= \frac{q\left(\pi\rho\,(2i-1)\,r^2 - (8i-4)\right)}{\pi\rho\,(2i-1)\,r^2}\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right). \end{aligned} \tag{5}$$

As for the ith ring with a probability p_i to generate interference route, the average nodal energy consumption in intracluster communication in each data collection period can be calculated as $E_i^{\text{in}} = e_i^{\text{in}} p_i$.

(2) Energy consumption in cyclic diversionary routing: when the ith ring generates the cyclic diversionary routing, every cluster head in this ring will send and receive data for one time. And the data transmission distance can be defined as twice of the cluster radius; then here it is r. So the average energy consumption in the ith ring with interference route to generate is $e_i^{cr} = N_i^c(E_t^{\sigma,r} + E_r^{\sigma})/N_i = (8i-4)(E_t^{\sigma,r} + E_r^{\sigma})/\pi\rho(2i-1)r^2$. Combining with the probability p_i, the energy consumption in cyclic diversionary routing of the ith ring in one data period is $E_i^{cr} = e_i^{cr} p_i$.

No matter whether the ith ring has to generate the interference route or not, one cluster head has to forward data to the cluster head in the inner ring to create the main routing to the sink. The transmission distance can be defined as r and this energy consumption is $E_i^{\text{backbone}} = (E_t^{\sigma,r} + E_r^{\sigma})/N_i = (E_t^{\sigma,r} + E_r^{\sigma})/\pi\rho(2i-1)r^2$.

In conclusion, the energy consumption of the ith ring ($i \neq 1$) can be calculated as the following formula:

$$\begin{aligned} E_i^{\text{avg}} &= E_c + E_i^{\text{in}} + E_i^{cr} + E_i^{\text{backbone}} \\ &= E_c + \frac{p_i q\left(\pi\rho\,(2i-1)\,r^2 - (8i-4)\right)}{\pi\rho\,(2i-1)\,r^2}\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right) \\ &\quad + \frac{p_i\,(8i-4)\left(E_t^{\sigma,r} + E_r^{\sigma}\right)}{\pi\rho\,(2i-1)\,r^2} + \frac{E_t^{\sigma,r} + E_r^{\sigma}}{\pi\rho\,(2i-1)\,r^2}. \end{aligned} \tag{6}$$

□

From our routing strategy, we can see that the network security rises with greater probability p_i of generating the interference route in a ring. To achieve the best situation with network security and maximal lifetime, we must balance the energy consumption of all the nodes in the network. In this case, the remaining energy in some nodes will be fully used to generate interference routes and die at the same time with hotspots who decide the lifetime of the network.

Theorem 2. *In order to furthest improve the location privacy security without affecting the network lifetime, the nodal energy consumption of the ith ring E_i^{avg} should meet the following constraints, where m means the number of rings in the network:*

$$\begin{gathered} E_i^{\text{avg}} = E_{i+1}^{\text{avg}} \mid i \in \{2, \ldots, m-1\}, \\ E_i^{\text{avg}} \leq E_1^{\text{avg}} \mid i \in \{2, \ldots, m\}. \end{gathered} \tag{7}$$

Proof. First, to ensure no effect on the network lifetime, the area that consumes the most energy must be the hotspot area, which consumes $E_{\max} = E_1^{\text{avg}} = e_u/2\pi r$. Meanwhile, the average nodal energy consumption of the ith ring should meet the condition $E_i^{\text{avg}} \leq E_{\max}$, which is

$$\begin{aligned} &E_c + \frac{p_i q\left(N_i - N_i^c\right)\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right) + p_i N_i^c\left(E_t^{\sigma,r} + E_r^{\sigma}\right) + E_t^{\sigma,r} + E_r^{\sigma}}{N_i} \\ &\quad \leq \frac{E_t^{\sigma,r/2} + E_r^{\sigma}}{\pi r^2 \rho} \mid i \in \{2, \ldots, m\}. \end{aligned} \tag{8}$$

Then, to furthest improve the source-location privacy security, we must keep a balance between the average nodal energy consumption E_i^{avg} of nodes in each ring. Thus, we have $E_i^{\text{avg}} = E_{i+1}^{\text{avg}} \mid i \in \{2, \ldots, m-1\}$, which is

$$\begin{aligned} &\frac{p_2 q \left(N_2 - N_2^c\right)\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right) + p_i N_2^c \left(E_t^{\sigma,r} + E_r^{\sigma}\right) + E_t^{\sigma,r} + E_r^{\sigma}}{N_2} \\ &= \frac{p_3 q \left(N_3 - N_3^c\right)\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right) + p_3 N_3^c \left(E_t^{\sigma,r} + E_r^{\sigma}\right) + E_t^{\sigma,r} + E_r^{\sigma}}{N_2} \\ &= \cdots = \frac{p_m q \left(N_m - N_m^c\right)\left(E_t^{\sigma,r/2} + E_r^{\sigma}\right)}{N_m} \\ &\quad + \frac{p_m N_m^c \left(E_t^{\sigma,r} + E_r^{\sigma}\right) + E_t^{\sigma,r} + E_r^{\sigma}}{N_m}. \end{aligned} \tag{9}$$

Once we integrate the two above constraints, Theorem 2 can be proved. □

Inference 1. The network lifetime in this work, where several extra cyclic routes are created, is the same as that of the shortest path routing.

Proof. If the energy consumptions of other rings are no greater than the first ring, the network lifetime is determined by the first ring who has the highest level of energy consumption. If the energy consumption of the first ring in these two strategies are the same, then the network lifetime are the same. As we can see from the previous discussion, only one route to the sink is created, and with the strategy in this work, several cyclic diversionary routes are created, but still there is only one route in the first ring. Besides, even with diversionary routes created in other rings, according to Theorems 1 and 2, the energy consumption of other rings will not be greater than the first ring, so the network lifetime of these two strategies remains the same. □

5. Performance Analysis

In this section, we will analyze the security of the proposed cyclic diversionary routing scheme. Then through network security criteria and lifetime criteria, we figure out the probability p_i to generate interference route in each ring and the probability q of cluster members to send dummy messages in that ring. Finally, we make an evaluation of the transmission delay of the network. From the following analysis, we can see that our scheme brings a better network security and maximal network lifetime along with longer transmission delay, which means that our scheme is best suited in applications where certain amount of data latency is tolerated.

5.1. Security Analysis. For the powerful detection ability of global eavesdroppers, we must generate several fake events to confuse the adversaries at random time. The numbers of these events can well speak of the network security. Since the adversaries cannot distinguish the real events from all the events that occur in the network, we can achieve a better network security with more homogeneous events at certain time.

Theorem 3. *In CDR scheme, the security of location privacy against global eavesdroppers has been improved S_{CDR} times, and S_{CDR} meets the following equation:*

$$\begin{aligned} S_{\text{CDR}} = \sum_{2}^{m} N_i^{\text{cfs}} p_i &= \left(\pi\rho r^2 + 4 - 4q\right)\left(m^2 + m - 2\right) \\ &\quad + (m-1)\left(4q - \pi\rho r^2 - 4\right). \end{aligned} \tag{10}$$

Proof. From the establishment process of CDR, we can see only 3 steps involved in data transmission. Since the main routings are generated in every period and all start from the outer ring to the inner ring in the shortest path, adversaries can find nothing useful from the main routings. And also in cyclic diversionary routes, we apply the same rules in generating interference route and all the cyclic diversionary routes are homogeneous.

In this case, all the cluster members that send dummy messages and the cluster heads in rings that generate the interference routes in CDR scheme can effectively confuse the adversaries in source-location analysis. In other words, numbers of all the cluster members that send dummy messages and the cluster heads in those rings can be seen as security enhancement levels in our CDR scheme.

In one data aggregation period, if the ith ring generates the interference route, the numbers of cluster members involved in intracluster communication are $N_i^{cfs} = (N_i - N_i^c) \cdot q + N_i^c = \pi\rho(2i-1)r^2 + (1-q)(8i-4)$. Since the ith ring has the probability p_i to generate interference route, we can find that in one period CDR scheme can provide S_{CDR} times the security level than that of the greedy routing which has no network security protections:

$$\begin{aligned} S_{\text{CDR}} = \sum_{2}^{m} N_i^{cfs} p_i &= \left(\pi\rho r^2 + 4 - 4q\right)\left(m^2 + m - 2\right) \\ &\quad + (m-1)\left(4q - \pi\rho r^2 - 4\right). \end{aligned} \tag{11}$$

□

5.2. Parameters in CDR Scheme. From Theorems 2 and 3, the probabilities p_i and q both affect the network lifetime and network security. Here we meet an NP-hard problem to achieve the optimization of two parameters. To obtain the best security with the assurance of maximal lifetime, we propose the following algorithm to figure out p_i and q_i.

5.3. Delay Analysis. In this section, we will discuss the delay caused by the CDR scheme in detail. In CDR scheme, we cannot ensure the shortest delay of data transmission since we have much delay in intracluster communication and cyclic diversionary routing. With more interference routes and more cluster members to send dummy messages, we have better network security and also longer transmission delay.

From the first algorithm, we find that transmission delay gets maximum when all the rings generate interference routes other than the first ring, which also brings a maximal network security. Algorithm 2 gives a detailed method to figure out the probability pi. Then we can calculate the average transmission delay in CDR scheme.

Theorem 4. *Assume the average delay of data transmitted from a sensor to its neighbor sensor is d_u. Then, in CDR scheme, the delay of the real data message from being sent to being received by the sink is*

$$D_{\mathrm{CDR}} = \pi\rho\left(\frac{r}{2}\right)^2 d_u + \sum_{2}^{m} d_u p_i\,(8i-4) + \lceil R/r\rceil\, d_u. \quad (12)$$

Proof. Here we also divide the transmission delay into three parts.

(1) Delay in intracluster communication: according to our clustering algorithm, we use the TDMA method to conduct intracluster data aggregation. Since data package has an average transmission delay d_u in every hop, we can assign the time slot for each cluster member as d_u. The quantity of nodes in one cluster is $N_{cn} = \pi(r/2)^2 \cdot \rho$. So our delay in intracluster aggregation is the same as the overall time slots of the cluster members, which is $D_{\mathrm{in}}^{\mathrm{avg}} = N_{cn}d_u = \pi\rho(r/2)^2 d_u$.

(2) Delay in cyclic diversionary routing: we forward our data around cluster heads hop by hop in a ring that needs to generate interference route with probability p_i. The number of the cluster heads in the ith ring can be calculated from Theorem 1 as $N_i^c = 8i - 4$. So in the ith ring we have a transmission delay $D_i^{cr} = N_i^c d_u = d_u(8i-4)$. In this case, we can calculate the average delay in cyclic diversionary routing in one data aggregation period as $D_{cr}^{\mathrm{avg}} = \sum_2^m D_i^{cr} p_i = \sum_2^m d_u p_i (8i-4)$.

(3) Delay in backbone routing: according to our routing protocol, we generate a backbone route in every data aggregation period which starts from the outmost ring to the sink in greedy routing way. The number of rings in the network is $N_r = \lceil R/r \rceil$, so the delay in backbone routing can be expressed as $D_{br}^{\mathrm{avg}} = N_r d_u = \lceil R/r \rceil d_u$.

After integrating all the information above, we figure out our transmission delay D_{CDR}:

$$\begin{aligned} D_{\mathrm{CDR}} &= D_{\mathrm{in}}^{\mathrm{avg}} + D_{cr}^{\mathrm{avg}} + D_{br}^{\mathrm{avg}} \\ &= \pi\rho\left(\frac{r}{2}\right)^2 d_u + \sum_{2}^{m} d_u p_i\,(8i-4) + \left\lceil \frac{R}{r} \right\rceil d_u. \end{aligned} \quad (13)$$

□

6. Experimental Results

In this section, we conduct simulations to compare the performance of CDR scheme with TFS [17] and greedy routing scheme (GR, i.e., a scheme without any preservation measure, in which a sensor sends the real message to the sink once detecting an object). The simulation is based on OMNET++, which is an open network simulation platform for large network. In this simulation, sensors are randomly deployed in a circular area and the sink is located at the centre of the field. In the case of unspecified network parameters, $R = 500$ m, $r = 80$ m, $\rho = 0.002$, $\sigma = 1000$ bits, and cluster radius is half of the transmission range. Because TFS is more applicable to square networks, we set the TFS network in a $2R \times 2R$ square area. To ensure a similar network situation with GR and CDR, R, r, ρ, and σ keep the same values as before. According to the settings in [17], we set the tree level in TFS as two and divide the network into a number of cells. Given the balance consideration of both energy consumption and network security, a cell is randomly selected to generate real event messages and other 20% cells generate fake messages. For a better analysis of the network lifetime and data latency, we assume a period $P = 10$ h $= 36000$ s in which the network detects the object and sends data to the sink, and the data transmission delay from one node to another is $d_u = 2$ s. And in the TFS scheme, the buffer interval T_{proxy} is set as $5d_u$. Other settings of the network are shown in Table 1.

6.1. Experimental Results of Energy Consumption and Network Lifetime.

In Section 4, we made a detailed analysis of the energy consumption in CDR and worked out the energy constraint formula to ensure network security in the context of maximal network lifetime. In this subsection, we evaluate the simulation results of the energy consumption and network lifetime in CDR scheme.

Figure 4 shows the nodal energy consumption in different regions under GR scheme and CDR scheme. We draw this curve with the average data from 1000 times of data gathering. As shown in this figure, we can see that in GR scheme hotspots near the sink consume much more energy than other nodes, and after the peak in energy consumption, nodes consume lesser energy as they are further away from the sink. Particularly in regions near the boundaries of the network, nodal energy consumption approaches to zero. However, in CDR scheme, nodal energy consumption shows a different trend. Hotspots near the sink still consume much energy and after reaching the peak, the energy consumption cure falls a little. Then with the increase in distance from the sink, nodal energy consumption level gradually rises and is maintained with a slight fluctuation below the peak in the end, which shows that with interference routings and looping strategy, CDR scheme successfully makes full use of the abundant energy in outer areas of the network and enhances the energy efficiency of the network.

Figures 5 and 6, respectively, show the 3D map of energy consumption in GR scheme and CDR scheme.

Figure 7 gives a detailed comparison of total energy consumption of the network between GR scheme and CDR scheme in one data aggregation period. From this figure, we can see that the whole network consumes much more energy in CDR than in GR scheme. We create many fake sources to increase network security with the abundant energy in outer areas of the network and these fake sources still consume more energy when they send fake messages. Figure 8 shows the tendency of different probabilities of interference routing generation in rings in relation to different probabilities of intracluster data aggregation. We can see that p and q have a

```
$p_i^{\text{optimal}} = 1$, $q^{\text{optimal}} = 0$, $S_{\max} = 0$
while $q < 1$ do
Figure out the nodal energy consumption of the outmost level $E_m^{\text{avg}}$ with Theorem 1;
According to Theorem 2, we can compute the maximum value of $p_m^q$ by the inequality of $E_m^{\text{avg}} \le E_1^{\text{avg}}$ and get the
    probability of establishing diversionary route in each level $p_i^q$ with $E_i^{\text{avg}} = E_m^{\text{avg}} \mid i \in \{2, \ldots, m\}$.
Work out the enhancement of location privacy security $S_q$ according to Theorem 3;
if $S_q < S_{\max}$ then
        $S_{\max} = S_q$;
        $p_i^{\text{optimal}} = p_i^q \mid i \in \{2, \ldots, m\}$;
        $q^{\text{optimal}} = q$;
    end if
$q = q + \varepsilon$, here $\varepsilon$ is a very small increment;
end while
End
```

Algorithm 2: Algorithm for figuring out p_i and q to get the strongest security performance.

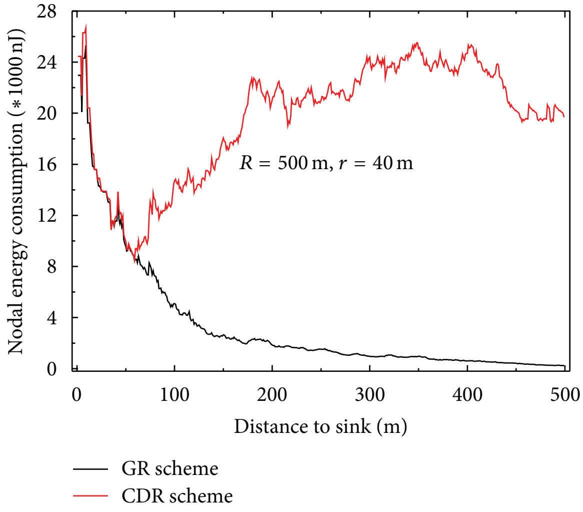


Figure 4: Energy consumption under different routes protocols.

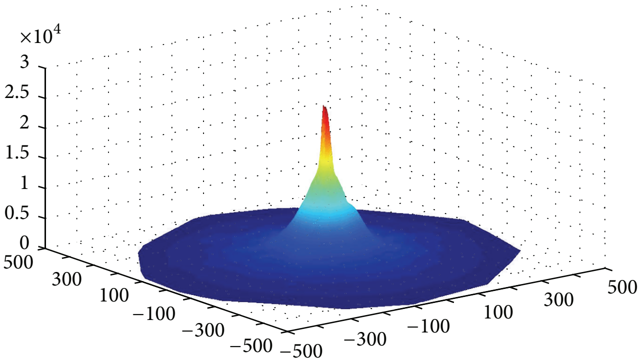


Figure 5: Energy consumption under GR scheme (3D).

negative correlation of each other. And with a specific value of q, rings in outer areas have higher probabilities in generating interference routings than inner rings.

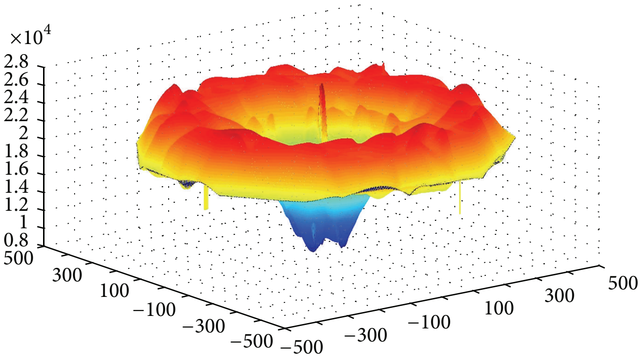


Figure 6: Energy consumption under CDR route scheme (3D).

Figures 9 and 10 compare the network lifetime of three protocols from different aspects. As shown in Figure 9, network lifetimes in GR and CDR schemes are basically the same with different transmission ranges while network lifetime in TFS is much shorter. We can find the explanation in Theorem 2. Since the network lifetime depends on the energy consumption of hotspots that are in one hop range of the sink, as long as we bring no extra energy consumption to these hotspots and keep the energy consumption of outer areas lower than this hotspot area, we can promise a maximal lifetime. Figure 10 gives the different network lifetimes in GR and CDR scheme with different network radii in different data aggregation rate. With a certain data aggregation rate, network lifetimes in GR and CDR nearly remain the same regardless of the change in network radii. With a certain network radius, network lifetime keeps rising along with the growing data aggregation period both in GR and in CDR schemes.

6.2. Experimental Results of the Security and Delay Performance. In the previous subsection, we analyze the energy consumption and network lifetime in CDR scheme and verify its energy efficiency. In this subsection, we test the security performance and transmission latency of this scheme. In

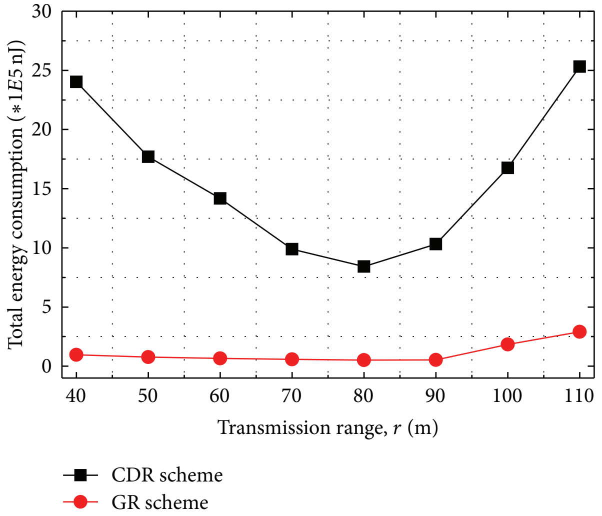


FIGURE 7: Comparison of total energy consumption of the network between GR and CDR.

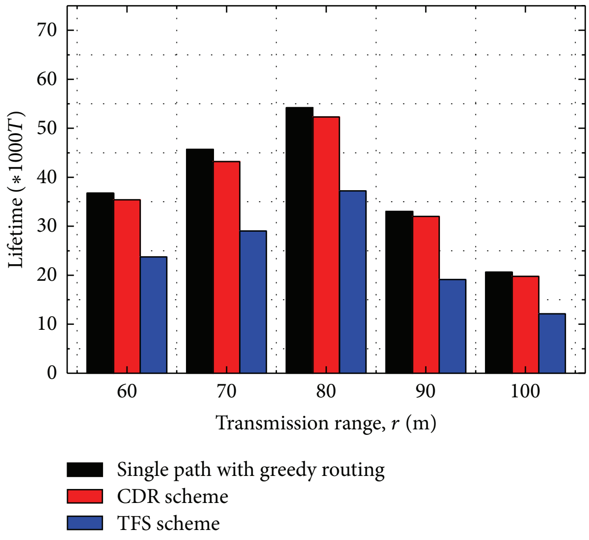


FIGURE 9: Network lifetime of different protocols with different transmission ranges.

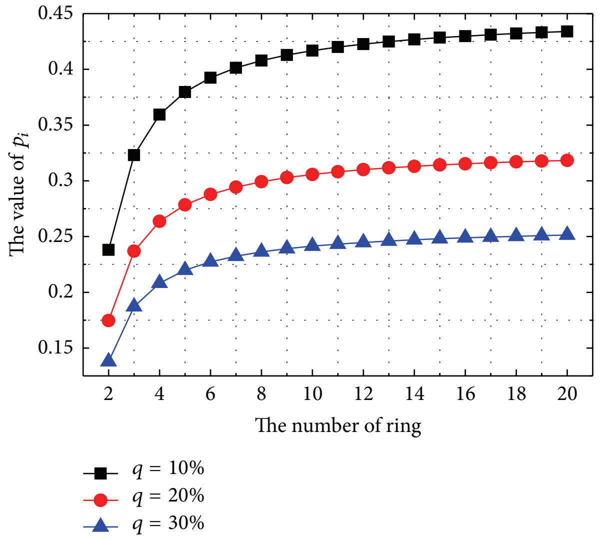


FIGURE 8: Different probabilities of interference routing generation in rings.

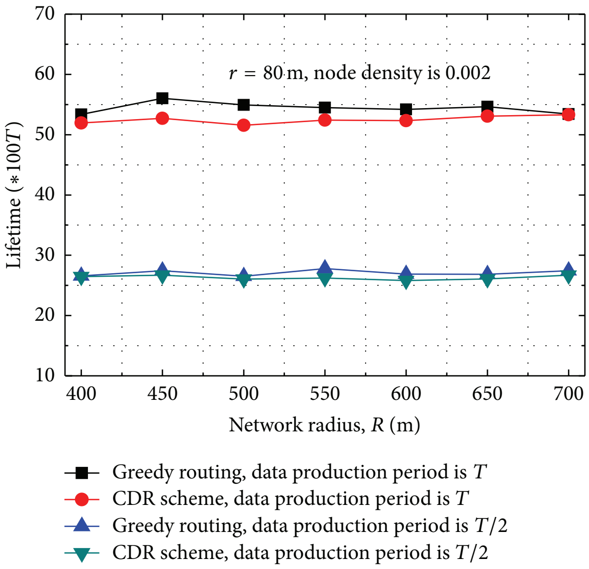


FIGURE 10: Different network lifetimes in GR and CDR.

our network model, the adversaries have a powerful ability in time correlation but cannot distinguish the authenticity of the data. In this case, we assume that all the nodes that perform intracluster communication in one period are the fake sources in the network. And we can treat the number of the fake sources as the criteria in network security evaluation.

Figure 11 compares the network transmission delay in three protocols. In GR scheme, once a real event is detected, the real source will directly forward this message to the sink in every period. And the transmission delay keeps rising as the distance grows between the sink and the event. In TFS scheme, proxies need time to process the fake messages which results in waiting delay and queuing delay. Therefore, transmission delay is a little longer in TFS scheme than in GR scheme. However, in CDR scheme, the transmission delay in one period of data aggregation is much longer than in both of the previous schemes. While with different locations of the real source in any part of the network, transmission delay maintains a minor fluctuation within certain boundaries. From Theorem 4 we can find that intracluster communication and intercluster communication both lead to transmission delay of the network, but the latency will not change much despite the different locations of the real source. Actually this feature can provide source-location privacy against the adversaries' ability in time correlation which will possibly help to locate the object. Figure 12 gives the relation between

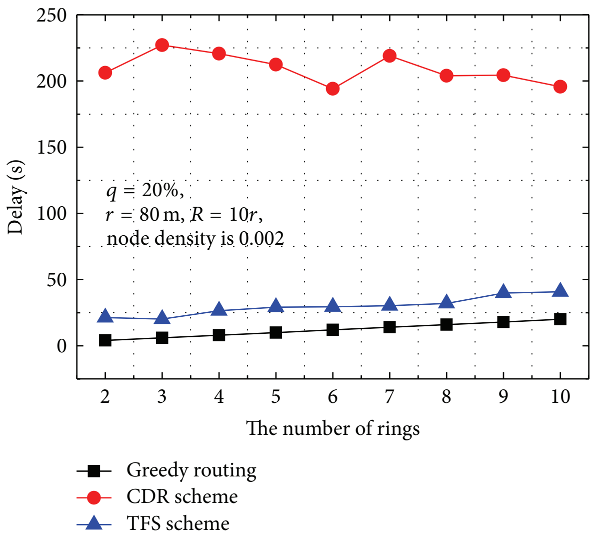


Figure 11: Delay comparison among different routing protocols.

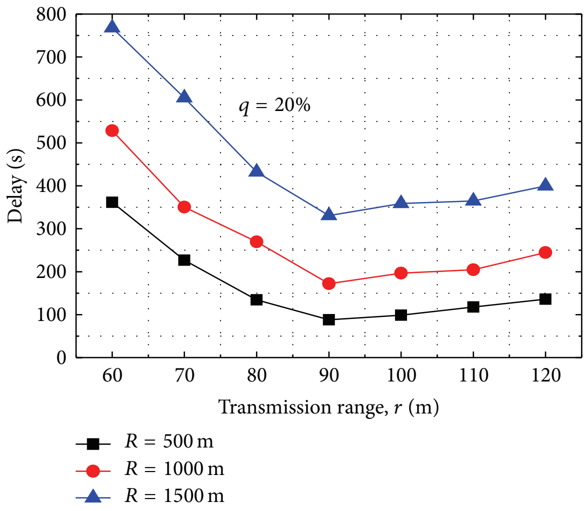


Figure 13: Delay in CDR scheme under different r.

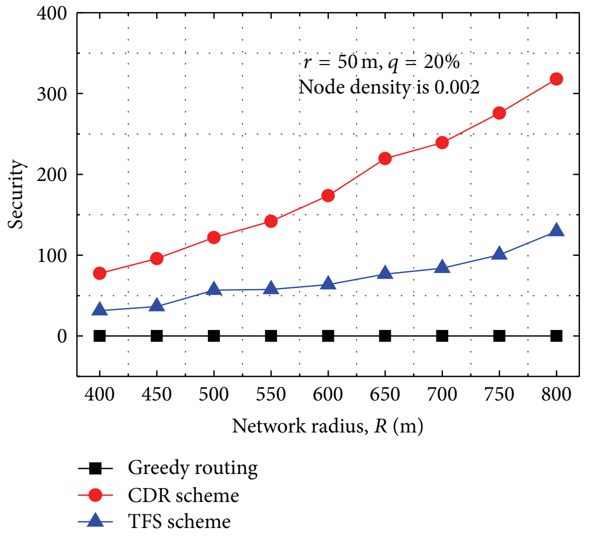


Figure 12: Security comparison among different routing protocols.

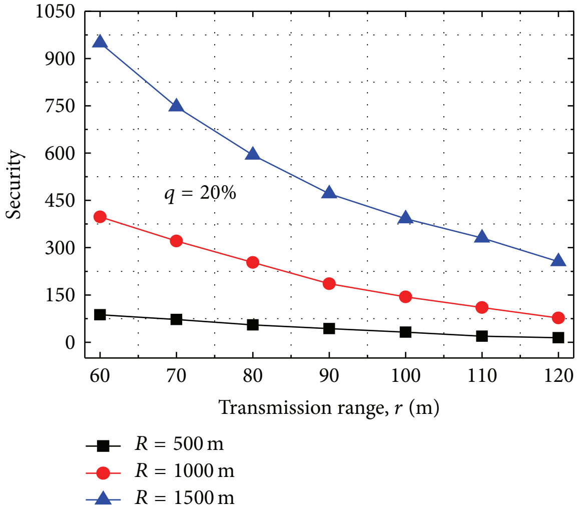


Figure 14: Security in CDR scheme under different r.

network security and network scale in GR, TFS, and CDR schemes. It can be seen that, in TFS scheme and CDR scheme, more fake sources are created to confuse the adversaries and bring more powerful security, and the security of the CDR scheme is much stronger than the others'. Moreover, as the scale of the network grows, the CDR scheme increasingly enhances the security of the network.

Figures 13 and 14 show the different transmission delays and network security with different transmission ranges in CDR scheme. In Figure 13, transmission delay drops as the transmission rate rises and it reaches the valley around r = 90 m. After that, it rebounds to rise as the r keeps rising. While with the same r, transmission delay exponentially multiplies as R keeps rising. All these results clearly indicate that network scale directly affects the transmission delay, and we can certainly find the proper r to achieve optimal data latency in a given network. In Figure 14, we can see that the network security drops as r rises and it rises in proportion as R rises with the same r. This shows that with smaller r, more rings are formed in the network and more interference routings are generated.

7. Conclusion

In this paper, we present an energy-efficient cyclic diversionary routing scheme based on clustering against global eavesdroppers in wireless sensor networks. The proposed scheme makes full use of the remained energy in regions far away from the sink to create cyclic diversionary routes as many as possible while with only one route in the region near the sink. This strategy improves the network security without sacrificing the network lifetime. Furthermore, we theoretically figure out the probabilities of different rings to generate interference routings and the probabilities that nodes send dummy messages. We find the optimal routing protocol to secure source-location with maximal network lifetime. Extensive performance analysis shows that the CDR scheme is better than the existing privacy preservation protocols.

Given the powerful abilities of the adversaries in network attack, our scheme is of special significance in source-location privacy preservation. While with an enhancement in network security, it also brings some data transmission delay to the network, which causes some influence in real-time applications. Therefore, in our future research, we will focus on the network latency optimization under the preconditions of a certain level of network lifetime and network security.

Acknowledgments

The authors are grateful to the anonymous reviewers for their industrious work and insightful comments. This work is supported by Mittal Innovation Project of Central South University under Grant no. 12MX15.

References

[1] H. Zhu, X. Lin, R. Lu, P. H. Ho, and X. Shen, "Slab: a secure localized authentication and billing scheme for wireless mesh networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 10, pp. 3858–3868, 2008.

[2] K. Ota, M. Dong, Z. Cheng, J. Wang, X. Li, and X. S. Shen, "Oracle: mobility control in wireless sensor and actor networks," *Computer Communications*, vol. 35, no. 9, pp. 1029–1037, 2012.

[3] A. Jhumka, M. Leeke, and S. Shrestha, "On the use of fake sources for source location privacy: trade-offs between energy and privacy," *Computer Journal*, vol. 54, no. 6, pp. 860–874, 2011.

[4] Y. Li and J. Ren, "Source-location privacy through dynamic routing in wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '10)*, San Diego, Calif, USA, March 2010.

[5] H. Wang, B. Sheng, and Q. Li, "Privacy-aware routing in sensor networks," *Computer Networks*, vol. 53, no. 9, pp. 1512–1529, 2009.

[6] H. Zhu, S. Du, Z. Gao, M. Dong, and Z. Cao, "A probabilistic misbehavior detection scheme towards efficient trust establishment in delay-tolerant networks," *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[7] A. Liu, J. Ren, X. Li, Z. Chen, and X. S. Shen, "Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks," *Computer Networks*, vol. 56, no. 7, pp. 1951–1967, 2012.

[8] A. Liu, X. Jin, G. Cui, and Z. Chen, "Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network," *Information Sciences*, vol. 230, pp. 197–226, 2013.

[9] A. Liu, Z. Liu, M. Nurudeen, X. Jin, and Z. Chen, "An elaborate chronological and spatial analysis of energy hole for wireless sensor networks," *Computer Standards & Interfaces*, vol. 35, no. 1, pp. 132–149, 2012.

[10] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[11] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382–401, 1982.

[12] A. Liu, Z. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multipath routing for WSNs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3255–3265.

[13] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS '05)*, pp. 599–608, June 2005.

[14] S. Armenia, G. Morabito, and S. Palazzo, "Analysis of location privacy/energy efficiency tradeoffs in wireless sensor networks," in *Proceedings of the Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet (NETWORKING '07)*, pp. 215–226, May 2007.

[15] Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon, "Entrapping adversaries for source protection in sensor networks," in *Proceedings of the International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM '06)*, pp. 23–32, IEEE Computer Society, Buffalo-Niagara Falls, NY, USA, June 2006.

[16] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor network routing," in *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '04)*, pp. 88–93, October 2004.

[17] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper," in *Proceedings of the 15th IEEE International Conference on Network Protocols (ICNP '07)*, pp. 314–323, Beijing, China, October 2007.

[18] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 77–88, April 2008.

[19] K. Bicakci, I. E. Bagci, and B. Tavli, "Lifetime bounds of wireless sensor networks preserving perfect sink unobservability," *IEEE Communications Letters*, vol. 15, no. 2, pp. 205–207, 2011.

[20] K. Bicakci, H. Gultekin, B. Tavli, and I. E. Bagci, "Maximizing lifetime of event-unobservable wireless sensor networks," *Computer Standards and Interfaces*, vol. 33, no. 4, pp. 401–410, 2011.

[21] S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin, and A. O. Fapojuwo, "A centralized energy-efficient routing protocol for wireless sensor networks," *IEEE Communications Magazine*, vol. 43, no. 3, pp. S8–S13, 2005.

[22] 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.

[23] V. Mhatre and C. Rosenberg, "Design guidelines for wireless sensor networks: communication, clustering and aggregation," *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.

[24] Z. Zhou, S. Zhou, S. Cui, and J. H. Cui, "Energy-efficient cooperative communication in a clustered wireless sensor network," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3618–3628, 2008.

[25] A. Förster, A. Förster, and A. L. Murphy, "Optimal cluster sizes for wireless sensor networks: an experimental analysis," *Ad Hoc Networks*, vol. 28, pp. 49–63, 2010.

[26] F. Wei, X. Zhang, H. Xiao, and A. Men, “A modified wireless token ring protocol for wireless sensor network,” in *Proceedings of the 2nd IEEE International Conference on Consumer Electronics, Communications and Networks (CECNet ’12)*, pp. 795–799, 2012.

[27] A. Ray and D. De, “Energy efficient cluster head selection in wireless sensor network,” in *Proceedings of the IEEE 1st International Conference on Recent Advances in Information Technology (RAIT ’12)*, pp. 306–311, 2012.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 532826, 11 pages
http://dx.doi.org/10.1155/2013/532826

Research Article

Energy-Efficiency of Cooperative Communication with Guaranteed E2E Reliability in WSNs

Deyu Zhang and Zhigang Chen

School of Information Science and Engineering, Central South University, Changsha 410083, China

Correspondence should be addressed to Zhigang Chen; czg@mail.csu.edu.cn

Received 24 December 2012; Revised 12 March 2013; Accepted 12 March 2013

Academic Editor: Mianxiong Dong

This paper addresses the energy efficiency of cooperative communication in WSN. We first establish the energy model of single-hop WSN. It is found that the cooperative communication is more suitable for harsh transmission environment with long-haul distance. The energy consumption per bit is numerically minimized by finding the optimal broadcasting BER and the number of cooperative nodes. Then, we expand the conclusion to the multihop scenario where "energy hole" dominates the longevity of WSN. To mitigate the energy consumption in the hotspots, as well as to keep the promised reliability, we adjust the transmission BER of the clusters according to the hops between the sink and cluster. On one hand, the statistical reliability is met. On the other hand, the energy consumed is converted from the nearer cluster (from the sink) to the farther ones. The network lifetime is thus optimized.

1. Introduction

WSN (Wireless Sensor Network), an energy-constrained network, has nodes mainly powered by batteries which are hard to replace even if possible. Numerous applications of WSN, such as environment monitoring, always need the network to operate for years without exchange of power suppliers. The prolongation of network lifetime is hence a critical design consideration and the data transmission must be energy efficient. More specially, the sensors near the sink are likely to die earlier since they are burdened with higher data load. Their deaths lead to the dysfunction of the network with the residual energy in the outside nodes. This is the well-known "energy-hole" phenomenon, the core of many researches in the literature [1].

MIMO (multiple-input and multiple-output) explores the spatial diversity of the wireless channel which can dramatically increase the channel capacity as well as the reliability of transmission. Once the transmission distance reaches a certain threshold [2], the energy conversation performance of MIMO systems can remarkably exceed the SISO (single-input-single-output) systems under the same Signal-to-Noise Ratio (SNR). The MIMO energy-efficiency transmission scheme is particularly useful for WSN due to the limited energy supplied. However, the direct application of multiple antennas technique on WSN is impractical for the insufficient physical size of sensor nodes. Fortunately, several individual sensors can cooperate for the data transmission in order to set up a Cooperative MIMO or MISO scheme, which are also known as Cooperative Communication (CC) [3].

CC scheme explores the energy efficiency of multiantennas technique which plays a significant role in the long-range transmission, where the transmission energy consumption dominates in the overall cost rather than that of the circuit [4]. Nonetheless, the decline of transmission energy consumption does not directly lead to the prolongation of network lifetime owing to the existence of "energy hole" [5]. The residual energy in the farther nodes may be up to 50% when the network dies [6]. Thus, the energy consumption balance is also the critical topic in the design of transmission scheme. In this paper, we first propose singleHop Algorithm for the minimization of energy consumption in single-hop scenario (see Algorithm 1). Furthermore, we generalize the conclusion to the multihop scenario and present the MultiHop Algorithm to mitigate the "energy hole" by adjusting the bit error rate (BER) at each cluster (see Algorithm 2).

Summarily, the main contributions of this paper are twofold.

```
Require: Network parameters, the maximum number of cooperative candidates $N_{CN}^{\max}$,
         transmission distance $d$ and maximum BER $p_e = 1 - \delta$.
Ensure: The optimal cooperative nodes $N_{CN}^*$, BER of the broadcasting $p_b^*$
         and the minimum energy consumption $E_{\min}$.
$E_{\text{circuit}} = \infty$, $E_{\min} = \infty$;
while ($N_{CN} < N_{CN}^{\max}$ and $E_{\text{circuit}} < E_{\min}$) do
    Compute the broadcasting radius $r_b$ according to (9);
    Calculate $p_b$ by (15) and decide $p_c$ by (13);
    Calculate the total energy consumption $E_{\text{tot}}$ by (17) and $E_{\text{circuit}}$ by (12);
    if ($E_{\min} > E_{\text{tot}}$) then
        $E_{\min} = E_{\text{tot}}$, $N_{CN}^* = N_{CN}$, $p_b^* = p_b$;
    end if
    $N_{CN} = N_{CN} + 1$;
end while
Output $N_{CN}^*$, $p_b^*$ and $E_{\min}$;
```

Algorithm 1: SingleHop Algorithm.

```
Require: Network parameters and the initial energy per node $E$,
         statistical reliability $\delta_t$.
Ensure: The optimal transmission BER sequence $p_e$.
Calculate the BER sequence according to Formula (31),
    record as $p_e^j = p_{e,1}^j, p_{e,2}^j, \ldots, p_{e,j}^j$, $j = 1, 2, \ldots n$;
Compute the optimal $N_{CN}^*$ and $p_b^*$ for each cluster by SingleHop algorithm,
    then calculate the average energy consumption of the nodes in each cluster by (28);
while can do
    Find the maximum $E_{\text{ave},a}$ at cluster $a$ and the minimum $E_{\text{ave},b}$ at cluster $b$;
    Find the max decreased $\Delta p_{e,a}^j$, and inclined $\Delta p_{e,b}^j$ to ensure the reliability
        can be met as well as keep the energy consumption of cluster $b$ slightly less than $E_{\text{ave},a}$;
    $p_{e,a}^j = p_{e,a}^j + \Delta p_{e,j}$, $p_{e,b}^j = p_{e,b}^j + \Delta p_{e,b}^j$;
end while
Output $p_e^j = p_{e,1}^j, p_{e,2}^j, \ldots, p_{e,j}^j$, $j = 1, 2, \ldots n$;
```

Algorithm 2: MultiHop Algorithm.

(1) Compared to the single-input and single-output (referred to SISO henceforth) transmission, it is revealed in [2, 7] that CC can save energy when the transmission distance exceeds the certain bound. In addition to this, we find that cooperative communication is more suitable for the long-haul transmission with higher requirement of BER in the harsh communication environment (larger path-loss parameter and power density of noise). Then, we propose the SingleHop Algorithm to choose the number of the cooperative nodes and the value of broadcasting BER to optimize the total transmission energy cost.

(2) In a multihop network, the sensors closer to the sink are more likely to be exhausted earlier due to the heavier data load. Based on the analysis of the single-hop scenario, we propose the MultiHop algorithm to prolong the lifetime of cluster-based network subject to the requirement of statistical reliability. Our strategy adjusts the transmission BER higher at the clusters farther away from the sink than the inner ones. This enables the near-sink cluster to lose the requirements of reliability. On one hand, the overall requirement can be met. On the other hand, the energy consumption of the near-sink clusters is shifted to the farther clusters.

The rest of this paper is organized as follows. The related work is given in Section 2. Section 3 presents the analysis of the single-hop network with CC scheme and SingleHop Algorithm. The numerical and experimental results are shown in Section 4. We further evaluate the energy consumption performance in a multihop clustered network, and Multihop algorithm is presented to mitigate the "energy hole" by adjusting the transmission BER in Section 5. Section 7 concludes the paper.

2. Related Work

A certain amount of research has recently been done to investigate various cooperative communication schemes. The author of [8] analyzed the performance of cooperative ARQ (automatic re-request) in both simple and hybrid schemes. It is pointed out that the cooperative ARQ protocols perform better than the traditional counterparts, even when the relay-destination channel is not as good as the source-destination channel, due to the spatial diversity explored by the cooperative protocols. Ikki and Ahmed investigated the capability of incremental-relaying mechanism for both decode-and-forward and amplify-and-forward relay schemes in [9]. Meanwhile, the closed-form expressions of BER and outage probability are proposed in their work. By the means of Alamouti space-time coding, Zhang et al. proposed a cooperative diversity system in [10], wherein the two users transmit data for each other, and the destination responds to the feedback at the middle of two Alamouti codes. To apply the distributed space-time codes in practice, the code distribution need to assign code matrix columns to individual cooperating nodes. Nonetheless, the basic setup in [8] and [10] includes only one intermediate relay node. As indicated in our work, more than 2 relay nodes may be demanded to optimize the transmission energy consumption.

From the perspective of energy consumption minimization, Cui et al. studied the characteristics of cooperative communication in WSN [2]. It is addressed that virtual multiple antennas are suitable for long distance transmission due to the extra circuit energy depletion. Based on this, Jayaweera studied the impact of the training overhead required in MIMO-based system and refined the conclusions obtained in [2]. However, the authors only consider the performance of cooperative transmission in comparison to the SISO systems. We generalize the object to the whole procedure of cooperative communication in cluster network (intracluster and intercluster) in our work. In [11], Li et al. analyze the energy consumption per unit transmit distance to achieve energy-efficient transmission. And the optimal transmission distance is obtained by turning the problem into a convex optimization problem. Nonetheless, the broadcasting BER is neglected in his work.

The selection of the "best relay" is applicable in case the source knows the CSI (channel statement information). In [12], the relay node selection and the transmission energy allocation are both studied based on the channel estimation at the source. This is implemented by the exchange of RTS/CTS messages. However, CSIR (channel statement information at the receiver), the analysis background of our paper, is more common for wireless link. Otherwise, the mature water-filling method can directly bring the optimal energy allocation scheme [13].

In [7], Zhang et al. analyzed the transmission distance in combination with the number of cooperative nodes. Then, the conclusion is extended to multihop scenario, as in our work. Hence, the optimal data transmission distance in each hop is obtained. Nevertheless, the authors merely consider the data gathering of the source node in [7]. Actually, the sensors in the network are all responsible for data collection, this is the fundamental reason for "energy hole" [14]. The global data gathering is analyzed for the rectangular scenario by Huang et al. in [15], wherein the network longevity is optimized by adjusting the cluster size. However, the authors omitted the analysis of parameters that significantly impact the network performance, especially the number of cooperative nodes and the reliability requirement. In [16], a clustered cooperative MIMO scheme based on LEACH is proposed by Yuan et al. wherein the authors concretely studied the operation process of the cluster construction. Unfortunately, the analysis of the influences of reliability and the number of cooperative nodes in cooperative communication are also ignored. In [17], Ota et al. proposed the actors' mobility control scheme in wireless sensor and actor networks (WSAN). By reinforcement learning in Markov decision processes, the energy efficient data collection scheme is addressed.

Table 1: Network parameters.

P_{Ct}	Power consumption of Tx circuits	98.2 mW
P_{Cr}	Power consumption of Rx circuits	112.5 mW
R_b	Transmission bit rate	10 kbps
ρ	The density of sensors in the network	0.1 perm^2
N_0	Thermal noise PSD	−171 dbm/Hz
E_F	The energy for data fusion per bit	5 nJ/bit
C	Communication constants	3.47×10^8

Table 2: Notations.

N_{CN}	The number of nodes participate in the cooperative transmission
r_b	The radius of broadcasting
p_e	The bit error rate (BER) induced by cooperative communication (in single-hop scenario, this is the end to end BER)
p_g, p_b, p_c	Bit error rate in data gathering, broadcasting, and cooperative transmission phase, respectively
R_C	The radius of clusters
$E_{\rm CH}$	The energy consumption of the cluster head in one round
$E_{\rm CPN}$	The energy consumption of the plain nodes participate in CC in one round
E_N	The energy consumption of the plain nodes in one round
D_i	The data amount sourced from cluster i
k	Path-loss exponent

3. Single-Hop System Description and Analysis

Table 1 presents the network parameters and the value of them. And for the convenience of readers to understand this paper, Table 2 summarizes the notations used in this paper.

3.1. System Model. We first introduce CC in a single-hop scenario, as seen in Figure 1. The relay node (particularly the cluster heads) broadcasts the data to its neighbors. The candidate nodes covered by the broadcasting would participate in the following CC phase, wherein the relay node and the cooperative nodes transmit the data simultaneously encoded by STBC [18] (space-time block coding) to the next relay node (or sink). This procedure of CC can also be seen in [19].

The energy consumption of the circuit blocks, except the power amplifier, for the transmission and reception of data packet, is summarized to constants represented by P_{Ct}

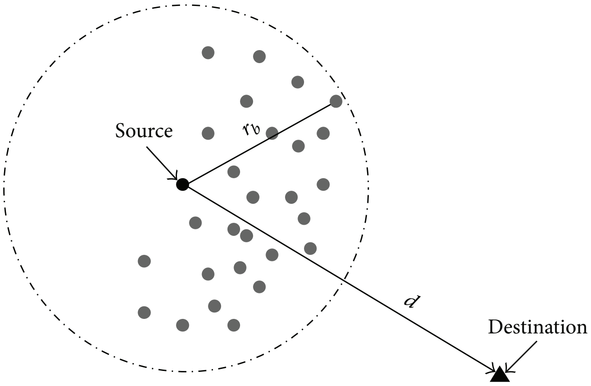


Figure 1: Impact of transmission range on total energy consumption.

and P_{Cr}. The power consumption of the amplifier can be approximated as follows:

$$P_{\text{Amp}} = (1+\alpha) P_s, \tag{1}$$

where $\alpha = \xi/\eta - 1$ with ξ the peak-to-average ratio and η the drain efficiency of the RF power amplifier.

According to [15], the energy consumption of one participated node in the cooperative transmission phase can be expressed as follows:

$$P_s = C\frac{E_b R_b d^k}{N_{CN}}, \tag{2}$$

where E_b is the required energy per bit at the receiver for the demanded bit error rate (BER). R_b denotes the data rate in bit with STBC coding. N_{CN} represents the number of nodes participated in the cooperative transmission, including the relay node and candidates. C is the product of several constants defined by $C = (4\pi)^2 M_l N_f / G_T G_R \lambda^2$ [15], where G_T and G_R are the gains at the transmit and receive antennas. λ is the carrier wavelength, M_l denotes the link margin of RF amplifier, and N_f is the receiver noise figure.

Since α in (1) solely depends on the modulation scheme and the associated constellation size, and we use BPSK to modulate the signal with the same constellation size throughout this paper, for brevity, C is expanded to be

$$C = (1+\alpha)\frac{(4\pi)^2 M_l N_f}{G_T G_R \lambda^2} \tag{3}$$

as adopted in [7].

We assume the fading of channel satisfies Rayleigh distribution. According to [15], the relationship between the BER and the received energy at the receiver can be derived to be

$$E_b \le \frac{N_{CN} N_0}{p_e^{1/N_{CN}}}, \tag{4}$$

where N_0 denotes the single-sided thermal noise power density (PSD) at room temperature. By approximating the bound as equality as well as substituting the equality and (3) into (1), the energy consumption of the amplifier can be expressed as in [15]:

$$P_{\text{Amp}} = C\frac{N_0 R_b d^k}{p_e^{1/N_{CN}}}. \tag{5}$$

$p_e^{1/N_{CN}}$ is the required BER at the transmitter (hereafter, referred as T-BER).

Summarily, the total energy consumption of each node for a fixed data rate can be derived as in [15]:

$$E_T\left(d^k, p_e, N_{CN}\right) = C\cdot\frac{N_0 d^k}{p_e^{1/N_{CN}}} + \frac{P_{Ct}}{R_b}. \tag{6}$$

The power needed for reception of nodes per bit is

$$E_R = \frac{P_{Cr}}{R_b}. \tag{7}$$

3.2. The Energy Consumption of CC. The broadcasting radius of the relay node is r_b. The energy consumption for the broadcasting with BER and the reception of the candidates can be derived as

$$E_b\left(r_b^k, p_b, N_{CN}\right) = E_T\left(r_b^k, p_b, 1\right) + \left(N_{CN}-1\right)\cdot E_R. \tag{8}$$

Based on the fact that r_b is much less than the transmission distance d. The differences of BER between the candidates are omitted throughout this paper. The number of candidates covered by the broadcasting radius complies with

$$N_{CN}\left(r_b\right) = \left(\pi r_b^2\right)\rho. \tag{9}$$

After the broadcasting phase, the cooperative nodes and the relay node transmit the data to the destination with BER p_c, the total energy consumption in this phase is

$$E_{CT}\left(d^k, p_c, N_{CN}\right) = E_T\left(d^k, p_c, N_{CN}\right) + E_R. \tag{10}$$

Eventually, the energy consumption can be summarized to be

$$E_{\text{CC}} = E_b\left(N_{CN}, p_b\right) + E_{CT}\left(N_{CN}, p_c\right). \tag{11}$$

Notably, the energy consumption for the reception of the destination is included in (11). And the circuit power cost can be expressed by

$$E_{\text{circuit}} = N_{CN} E_R + \left(N_{CN}+1\right)\cdot\frac{P_{Ct}}{R_b}. \tag{12}$$

The BER at the destination is

$$p_e = 1 - \left(1-p_b\right)\left(1-p_c\right) \approx p_b + p_c. \tag{13}$$

The partial derivative of E_{CC} with respect to p_b is

$$\frac{\partial E_{\text{CC}}}{\partial p_b} = \left(\frac{d^k}{\left(p_e - p_b\right)^{(1/N_{CN}+1)}} - \frac{r_b^k}{p_b^2}\right)\cdot C\cdot N_0. \tag{14}$$

The minimum E_{CC} is obtained in the following case:

$$\frac{p_b^2}{(p_e - p_b)^{(1/N_{CN}+1)}} = \frac{r_b^k}{d^k}. \qquad (15)$$

It is proved that (15) has only one real solution in $p_b \in (0, p_e)$ in the Appendix. Although the closed-form solution of p_b is unsolvable, we can obtain the numerical solution to (15).

The corresponding energy consumption for SISO scheme with BER p_e is

$$E_{SISO}(d^k, p_e) = E_T(d^k, p_e, 1) + E_R. \qquad (16)$$

Summarily, the total energy consumption according to the transmission scheme can be expressed by

$$E_{tot}(N_{CN}, p_e) = \begin{cases} E_{CC}, & \text{if } N_{CN} > 1, \\ E_{SISO}, & \text{if } N_{CN} = 1. \end{cases} \qquad (17)$$

3.3. Cooperative Communication Energy Consumption Optimization. As shown in Section 4, the number of cooperative nodes N_{CN} and the broadcasting BER p_b have significant impact on the overall energy cost of data transmission. However, getting the optimal value of N_{CN} and p_b is very difficult due to the complexity of Formula (11). This paper proposed the algorithm of variables' selection for cooperative communication from the perspective of practice. We assume that the required reliability of CC is δ. Hence, the maximum BER is $p_e = 1 - \delta$.

It is worth noting that the circuit energy consumption increases linearly with the number of cooperative nodes, as shown in (12). In case $E_{circuit} > E_{min}$ happens, more cooperative nodes would only deteriorate the energy-efficiency performance. To reduce the calculating time, SingleHop Algorithm will finish immediately when the circuit energy consumption has exceeded the acquired minimum energy consumption. Obviously, we need to execute the algorithm only once in case the network settings and transmission distance are unchanged.

4. Numerical and Simulation Results of Single-Hop CC

The related network parameters are given in Table 1 if not specified. We use network simulator ns2 version 2.35 to conduct the simulations. For each data point in the figures, we run simulation on 20 randomly created networks and take the average.

Consistent with the results of [2, 7, 20], CC outperforms the SISO system when the transmission distance is beyond a certain threshold with low E2E BER ($p_e \approx 0.11\%$), as shown in Figure 2. And the crossover indicates where the energy saved by CC exceeds the extra circuit energy consumption in comparison with SISO system. Notably, we comprehensively consider the energy consumption of broadcasting and the reception in our model, which are omitted in [2, 7]. In addition to this, Figure 3 illustrates the proportions of the energy consumption of each operation in the total power consumed.

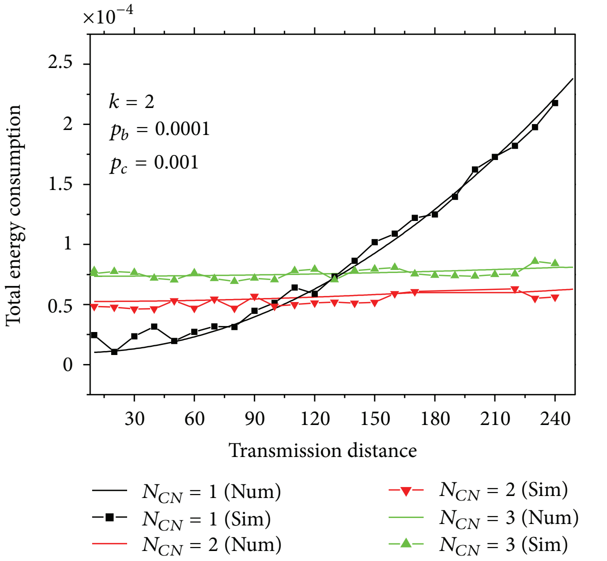


FIGURE 2: Transmission energy consumption per bit with low BER.

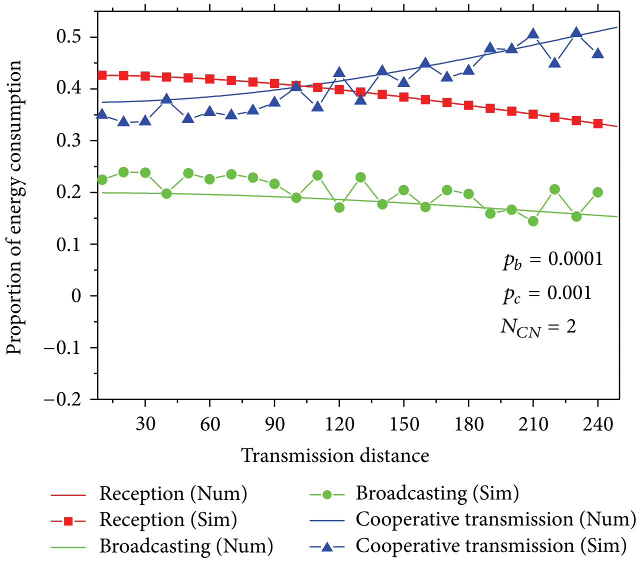


FIGURE 3: The proportional percentage of energy consumption.

Given that the energy expenditure of the data reception only depends on the number of cooperative nodes N_{CN}, the cooperative transmission takes a greater proportion as long as the transmission distance is sufficiently large ($d > 103$ m in Figure 6).

Figure 4 depicts the reason of the energy efficiency, where we plot the ratio of T-BER and the required E2E BER ($p_e^{1/N_{CN}}/p_e$) against the number of nodes participated in CC. The demanded T-BER $p_e^{1/N_{CN}}$ augments with the increasing number of cooperative nodes N_{CN}, as shown in Figure 4. The energy expenditure per node on cooperative transmission is eventually saved. Moreover, the derivative of $p_e^{1/N_{CN}}/p_e$ is with respect to p_e is $(1/N_{CN} - 1) \cdot p_e^{1/N_{CN}-1} < 0$; hence, $p_e^{1/N_{CN}}/p_e$ reversely related to the E2E BER p_e. Thus, the effect

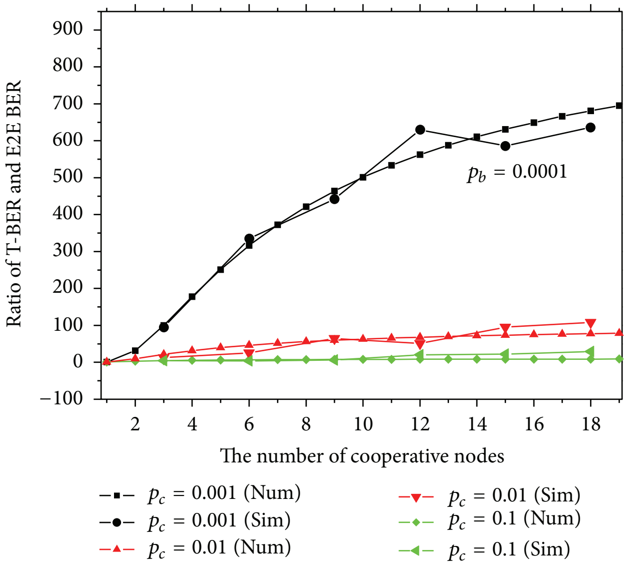


FIGURE 4: The ratio between end-to-end (E2E) BER and the T-BER.

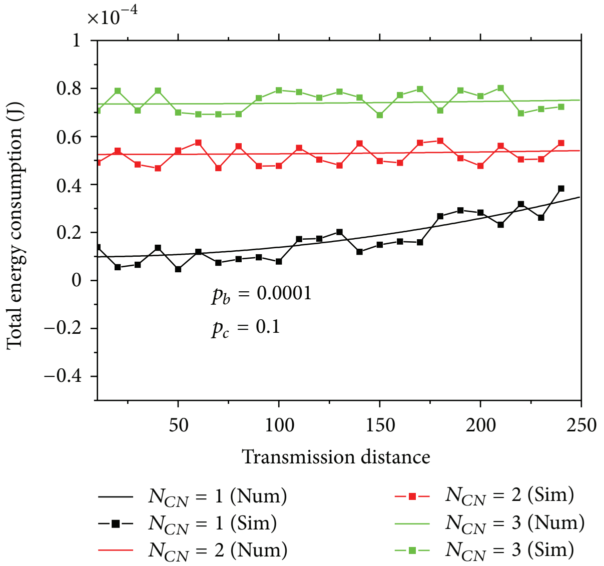


FIGURE 5: The comparison between CC and SISO scheme with high BER.

of CC is reduced by larger E2E BER. This explains why SISO system is always the optimal choice with low required E2E reliability ($p_e \approx 10\%$), illustrated in Figure 5.

We evaluate the performance of CC compared to SISO with path loss exponent $k = 2$ (in free space). Nevertheless, the transmitted signal would suffer the multipath fading ($k = 4$ when $d > 87$ [21]). As depicted in Figure 6, CC significantly outperforms the SISO system in multipath fading. In addition to this, the number of nodes that participate in CC relaxes the T-BER and further optimize the energy consumption performance with amply long transmission distance. Summarily, CC is more suitable for the longer transmission in harsh propagation environment (high path-loss exponent).

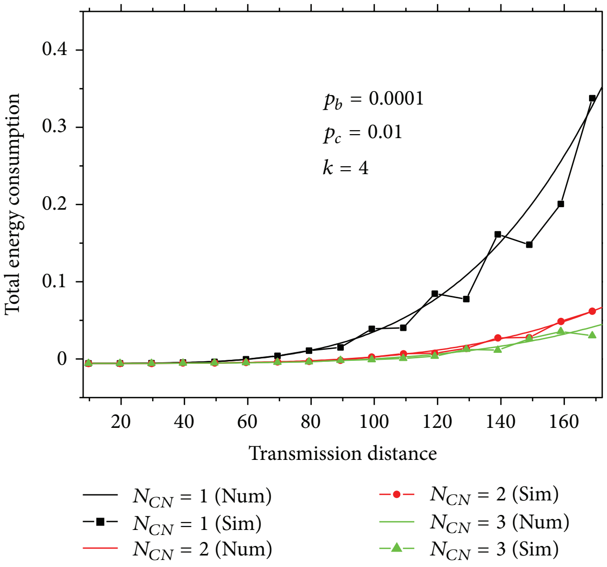


FIGURE 6: The energy consumption under multipath fading.

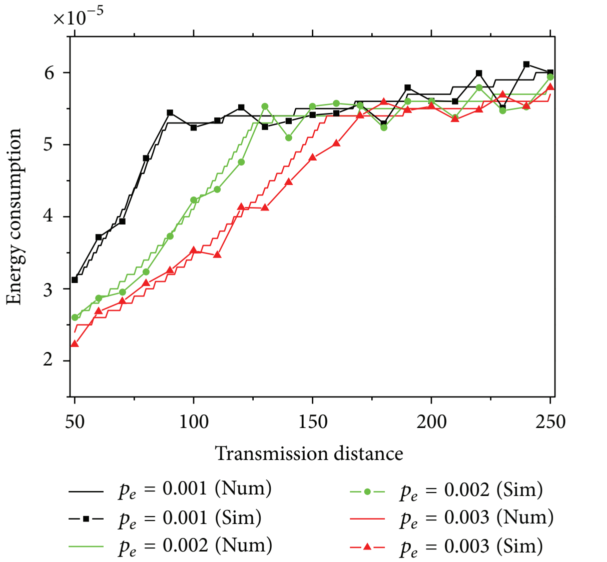


FIGURE 7: The optimal energy consumption by using SingleHop algorithm.

The performance of SingleHop algorithm is verified in Figures 7, 8, and 9. Take $p_e = 0.1\%$ as an example. CC is chosen when the transmission distance is beyond 103 m. Afterward, the rising trend of energy consumption remarkably declined compared to the SISO scheme due to the increasing of T-BER. The optimal number of plain nodes participated in CC is shown in Figure 8. When the number of cooperative nodes exceeds 1, CC is selected as the transmission scheme. Notably, since we take the average of multiple simulations, the number of nodes participate in CC may be decimals. Figure 9 plots the optimal broadcasting BER versus the transmission distance. The broadcasting BER takes only less than 2% in the whole BER, because the broadcasting

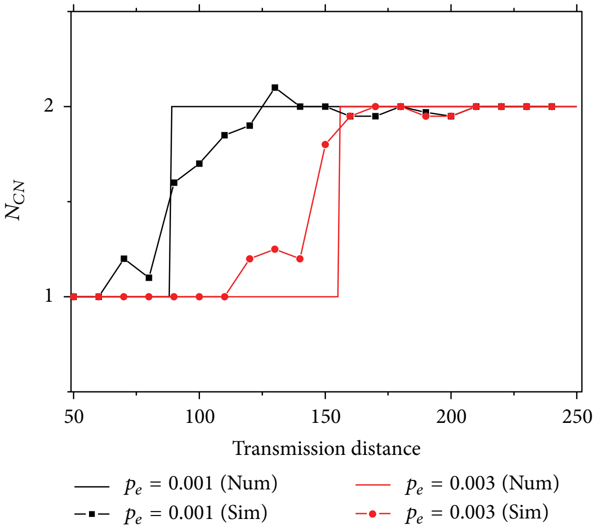


FIGURE 8: The optimal number of cooperative nodes.

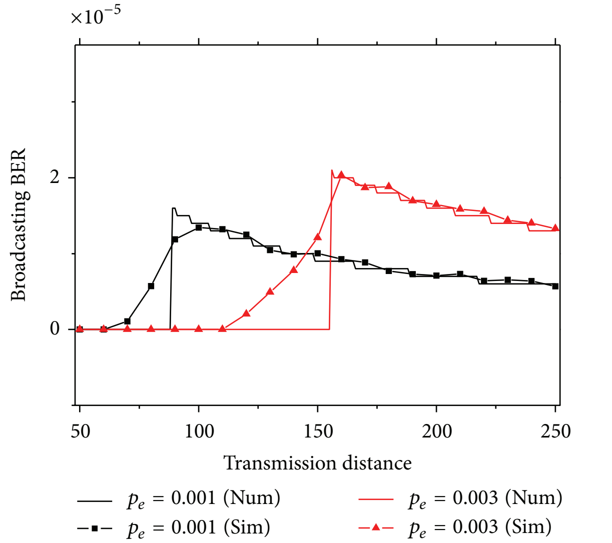


FIGURE 9: The optimal broadcasting BER obtained by SingleHop algorithm.

radius is much less than the transmission distance. As the transmit distance is growing, the reliability of broadcasting is even higher.

5. Maximization of Network Lifetime with Guaranteed E2E Reliability

In this section, we extended the conclusion of Section 3 to multihop scenario. As shown in Figure 10, nodes are densely dispersed in several circles which are far away from each other, and the clusters are linearly positioned [19]. The distance between the circles is much larger than the radius of those.

The radius of the ith clusters and the density of nodes are denoted by $R_{C,i}$ and ρ_i, respectively. The area of cluster i can be derived to be $S_i = \pi R_{C_i}^2$. $d_{C_i-C_{i-1}}$ denotes the distance between the ith and $(i-1)$th clusters and $d_{C_i-C_{i-1}} \gg R_{C_i}$. The channel fading satisfies Rayleigh distribution. And the path loss exponent is identical in both intra- and intercluster communication. The clusters are numbered by hops to sink.

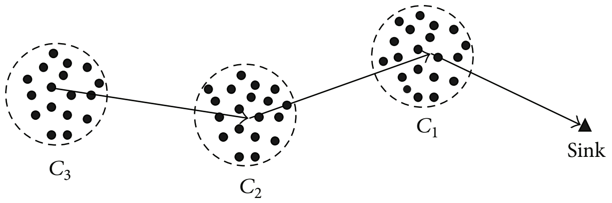


FIGURE 10: Multihop model.

5.1. Analysis of Energy Consumption and Bit Error Rate at Each Cluster. During the intracluster process, the plain nodes in cluster j transmit l bits data to the cluster head with BER $p_{g,j}$ in one round. Then, CH aggregates the data and chooses the transmission scheme based on SingleHop algorithm. If the cooperative communication is selected, CH broadcasts the data to the neighbors. The internal clusters are responsible for the relay of data stemming from outer clusters (C_2 and C_3 in Figure 10) in intercluster process. Notably, the notations in Section 3 are expanded in this section.

The BER in each step greatly influenced the energy consumption performance as we see in Section 3. Moreover, the overall BER consists of two parts, the BER at data gathering phase and the BER induced by the intercluster data transmission, respectively.

Here, we first investigate the relationship between BER in different phases and the required reliability. The overall reliability constraint is denoted by δ_t. δ_i^j represents the reliability for cluster i to transmit data stemmed from cluster j, and such a manner is employed in other notations. It is obtained apparently that $\delta_t = \prod_{k=1}^{i} \delta_k^j$.

Theorem 1. *To meet the overall required statistical reliability δ_t, the approximate accuracy of the data from cluster j is given by the following formula:*

$$p_{g,j} + \sum_{i=1}^{j} p_{e,i}^{j} \leqslant 1 - \delta_t. \tag{18}$$

Proof. At the jth cluster, (19) must hold

$$\left(1 - p_{g,j}\right)\left(1 - p_{e,j}\right) \leqslant \delta_j^j. \tag{19}$$

Analogy to the relationship of broadcasting BER p_b and cooperative transmission BER p_c is indicated in (13). We have (19) is approximated to be $p_{g,j} + p_{e,j} \leqslant 1 - \delta_j^j$. Expanding this procedure to following hops, we can acquire

$$\left(1 - p_{g,j}\right)\prod_{i=1}^{j}\left(1 - p_{e,i}^{j}\right)\prod_{i=1}^{j} \geqslant \delta_i^j, \tag{20}$$

which approximates the inequality (19). □

The nodes separately play 3 different characters in inter-cluster transmission, which are CH, cooperative nodes, and plain nodes, respectively. Based on the conclusion of Section 3, SingleHop algorithm is applied to determine the optimal value of $p_{b,i}^{j*}$, $p_{c,i}^{j*}$, and $N_{CN,i}^{j*}$. And the data load stemmed from cluster j is given by

$$D_j = \left(S_j\rho\right) \cdot l\varphi, \tag{21}$$

where φ is the fusion rate. And $E_{\mathrm{ag},j}$ denotes the energy consumption of data aggregation of cluster head (CH) in cluster j. Theorem 2 presents the analysis of energy consumption for each type of nodes:

Theorem 2. *$E_{CH\text{-}inter,i}^{j*}$ and $E_{CN\text{-}inter,i}^{j*}$ denote the power expenditure of the cluster head and the plain nodes which participate in CC during inter-cluster transmission. The energy consumption of the CH, cooperative nodes, and the plain nodes in cluster C_i are represented by $E_{CH,i}$, $E_{CH\text{-}inter,i}$, and $E_{N,i}$*

$$E_{CH,i}=\left(\sum_{j=i+1}^{n} D_j + \pi R_{C_i}^2 \cdot \rho\right) \cdot E_R+E_{ag,i}+\sum_{j=i}^{n}D_j \cdot E_{CH\text{-}inter,i}^{j*} \tag{22a}$$

$$E_{C\text{-}PN,i}^{j} = \begin{cases} D_j\left(E_R + E_{CN\text{-}inter,i}^{j*}\right), & if\ N_{CN,i}^{j*} > 1, \\ 0, & if\ N_{CN,i}^{j*} = 1, \end{cases} \tag{22b}$$

$$E_{N,i} = E_{g,i}\left(p_{g,i}\right). \tag{22c}$$

Proof. By CH rotation, any node in the cluster is able to be CH and the average distance between two randomly located nodes is $d_{\mathrm{ave},i} = 128R_{C_i}/(45\pi)$ [22]. Then, the energy consumption for each plain node can be expressed as follows:

$$E_{g,i} = E_T\left(d_{\mathrm{ave},i}^{k}, p_{g,i}, 1\right). \tag{23}$$

The energy consumption of CH for the data aggregation is

$$E_{\mathrm{ag},i} = D_i \cdot E_F, \tag{24}$$

where $E_F = 5\,\mathrm{nJ/bit}$ [15] is the power consumption of data fusion per bit. Set $E_{CT,i}^{j*}$ to denote the optimal energy consumption of inter-cluster transmission, namely, the output of SingleHop algorithm:

$$E_{CT,i}^{j*} = E_{\mathrm{tot}}\left(N_{CN,i}^{j*}, p_{e,i}^{j}\right). \tag{25}$$

In case CC is employed ($N_{CN,i}^{j*} > 1$), the energy consumption of CH is given by:

$$E_{CH\text{-}\mathrm{inter},i}^{j} = CN_0 \cdot \left[\frac{\left(d_{C_i-C_{i-1}}\right)^k}{\left(\mathrm{p}_{e,i}^{j} - p_{e,i}^{j*}\right)^{1/N_{CN,i}^{j*}}} + \frac{\left(r_{b,i}^{j}\right)^k}{p_{b,i}^{j*}}\right] + 2\frac{P_{Ct}}{R_b}. \tag{26}$$

The energy consumed by each plain node participated in CC is

$$E_{CN\text{-}\mathrm{inter},i}^{j} = CN_0\frac{\left(d_{C_i-C_{i-1}}\right)^k}{\left(p_{e,i}^{j} - p_{e,i}^{j*}\right)^{1/N_{CN,i}^{j*}}} + \frac{P_{Ct}}{R_b} + E_R. \tag{27}$$

The total data amount relayed by cluster C_i is $\sum_{j=i+1}^{n} D_j$. Hence we obtain (22a).

The energy cost of the cooperative nodes (except CH) on the reception of the data broadcasted by CH is $\sum_{j=i+1}^{n} D_j \cdot E_R$. So (22b) is acquired. □

In our paper, we assume that the CH and the cooperative nodes are selected based on the residual energy of the nodes. Therefore, it is considered that the energy consumption among the nodes are perfectly balanced, thus all nodes have approximate lifetime. Theorem 3 derives the average energy consumption of each clusters.

Theorem 3. *The average energy consumption per node in the ith cluster for an entire data gathering round is presented in the following:*

$$E_{ave,i} = \frac{E_{CH,i}}{N_i} + \sum_{j=i}^{n} E_{C\text{-}PN,i}^{j} \cdot \frac{N_{CN,i}^{j*} - 1}{N_{CN,i}^{j*}} + E_{N,i} \cdot \frac{N_i - 1}{N_i}, \tag{28}$$

where N_i denotes the number of nodes in C_i.

Proof. Nodes undertake the role of CH by cluster head rotation. Averagely, every node acts as CH for one time, as plain nodes for $N_i - N_{CN,i}^{j*}$ times after N_i data gathering round. In particular, the number of cooperative nodes depends on p_e^j, thus we consider the cooperative nodes in cluster separately according to the intercluster transmission scheme. Thus, (28) can be derived. □

Assume that the reliability $\delta_i^j = 1 - p_{e,i}^{j}$ is evenly distributed along the transmission trace. To meet δ_t, $p_{e,i}^{j}$ should satisfy

$$p_{e,i}^{j} \leqslant \frac{1 - \delta_t - p_{g,j}}{j}. \tag{29}$$

The network longevity optimization goal can be expressed as

$$\min_{0<i<n} \max E_{\mathrm{ave},i} \tag{30}$$

subject to

$$p_{g,j} + \sum_{i=1}^{j} p_{e,j} \leqslant 1 - \delta_t. \tag{31}$$

By applying this bound as equality, we obtain

$$p_{e,i}^{j} = \frac{\left(1 - \delta_t - p_{g,j}\right)}{j}. \tag{32}$$

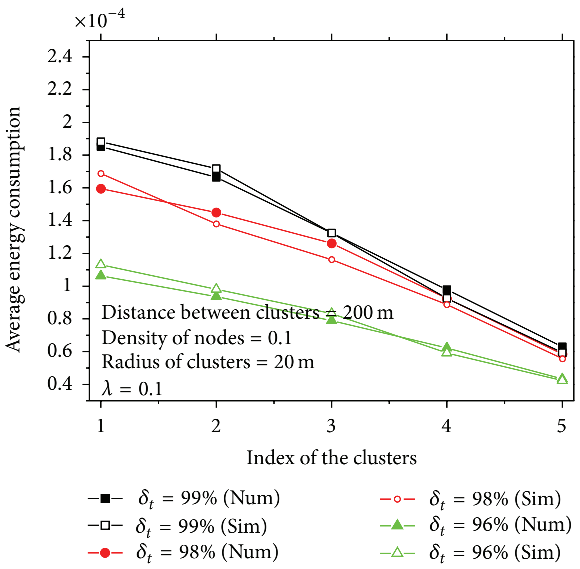


Figure 11: Average energy consumption of each cluster in one round.

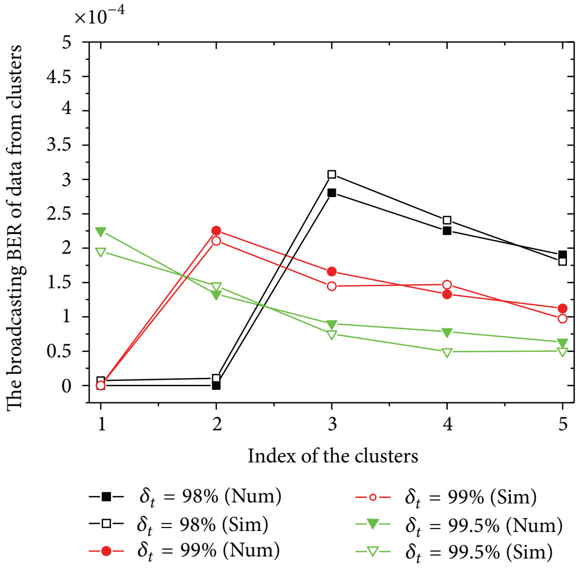


Figure 12: The broadcasting BER of data sourced from each cluster.

Notably, $p_{g,j}$ is known at the cluster head of cluster j. Thus, one can overhead $p_{g,j}$ into the data packet to inform the following clusters.

Recall that $d_{C_{i-1}-C_i} \ll R_{C_i}$, the BER induced by the inter-cluster transmission takes much higher proportion than that of intracluster. In this paper, it is set $p_{g,j} = (1-\delta_t/j)\cdot\tau$, where $0 < \tau < 1$ is a coefficient representing the proportion of p_g in the total BER. To make the analysis tractable and highlight the performance of CC in inter-cluster transmission, $\tau = 10\%$ is employed. And it is reasonable since the distance between the clusters is much larger than the radius of them. As we see in the proof of Theorem 2, the transport scheme of inter-cluster transmission depends on the required BER rather than the data amount. So we set the radius of the clusters identical to each other as $R_C = 20$ m. In addition to this, the impact of transmission distances on energy consumption is already stated in Section 3. And the distance between clusters are arranged to the same, $d_{C_i-C_{i-1}} = 200$ m.

We map the average energy consumption of each cluster in Figure 11. Obviously, cluster C_1 would die much earlier than the outside cluster because of the heavier burdened data load. This leads to the “energy hole” as well as the network paralysis [21]. Furthermore, in case the reliability of data along the transmission path is evenly distributed, the transmission scheme and broadcasting BER are also the same. Figure 12 depicts the optimal broadcasting BER for each cluster to transmit their own data. It is observed that SISO transmission is suitable for lower reliability transmission (clusters 1 and 2 when $\delta_t = 98\%$, cluster 1 when $\delta_t = 99\%$) while high-fidelity transmission prefers cooperative transmission. For instance, BER on each hop are almost (in spite the BER brought by the intra-cluster transmission) 2% and 0.4% for the clusters which are 1 hop and 5 hops to the sink according to the reliability 98%, respectively. SingleHop Algorithm selected the SISO scheme for cluster 1 and 2, where the broadcasting BER is zero, while CC is chosen for the peripheral clusters, as shown in the black lines in Figure 12.

6. Nodes Adopt the Different BER according to the Clusters They Belong to

Evidently, the cluster nearest to the sink dies much earlier than the clusters farther away which leads to “energy hole,” since the nodes in cluster 1 are burdened with larger amount of data. We notice that the reduction of power consumption at the energy hole leads to the prolongation of network lifetime. To mitigate this “energy hole” as well as maintain the statistical reliability, a strategy is proposed to convert the energy consumption at the energy hole to the farther part of the network by adjusting the transmission BER in each cluster. Based on the analysis in Theorem 3, the sum of BER along the routing path stays stable and the accuracy of the data can still reach the requirement of reliability. By means of this method, the energy consumption of the nearer clusters is reduced although the cost of the external clusters increased. As long as the maximum energy consumption declined, the network lifetime is optimized.

Through the calculation of MultiHop algorithm, Figure 13 plots the transmit BER of C_1 for the data from different clusters ($p_{e,1}^{j}$) compared to the originality. Since the energy expenditure of the clusters farther away from sink is lower than cluster C_1, BER for C_1, to transmit data is switched larger in order to balance the power cost. While to maintain the reliability, the BER of the farther cluster is relatively lower. Thus, the energy consumption of peripheral clusters increases and that of C_1 has declined as shown in Figure 14. Meanwhile, the longevity of network is improved (in case the initial energy of the nodes is $1J$, the lifetime is optimized by 9.85%).

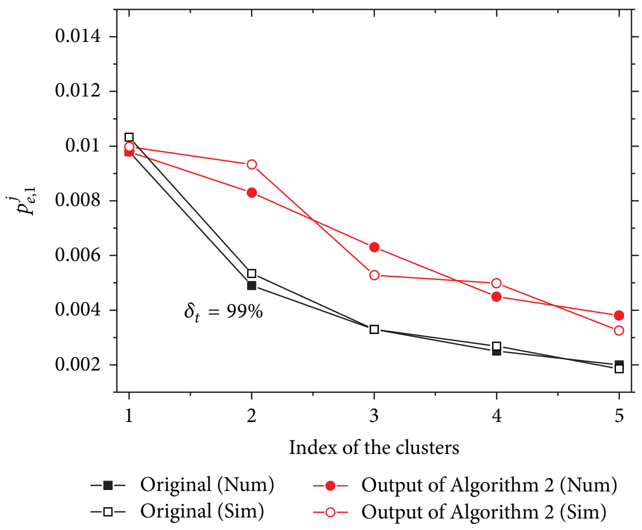


Figure 13: BER of transmission for cluster C_1.

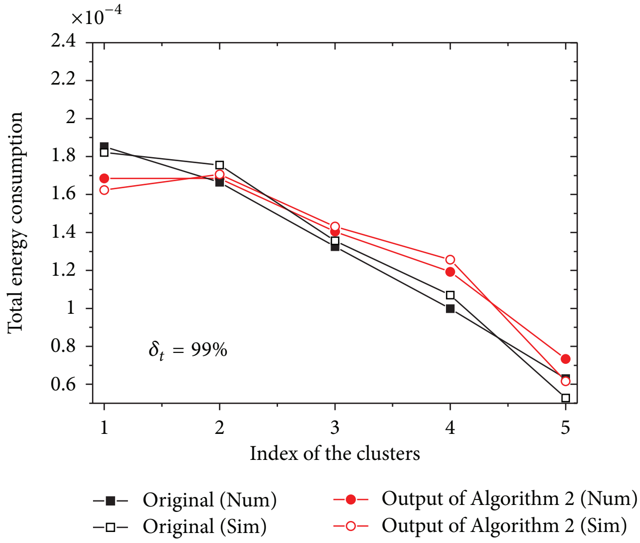


Figure 14: Energy consumption for one data gathering round.

7. Conclusion

In this paper, we jointly investigate the SISO and CC transmission schemes in both single-hop and multihop scenarios. The optimal number of cooperative nodes and the broadcasting BER are obtained for the energy efficiency. It is shown that cooperative communication is more suitable for the long-distance transmission in harsher environment. The conclusion of single-hop network is then expanded to multihop-clustered network where we study the energy cost of different nodes (cluster head, cooperative nodes, and plain nodes) in the cluster. Finally, we prolong the network lifetime by adjusting the transmit BER along the delivery path. An interesting extension is to precisely study the cooperative nodes selection scheme, since the probability is slightly different between nodes to be covered by broadcasting (the node at the core of the circle cluster is easier to be under the convergence).

Appendix

Suppose that $f(p_b) = p_b^2/(p_e - p_b)^{(1/N_{CN}+1)} - r_b^k/d^k$, we first prove that (15) has real solution when $p_b \in (0, p_e)$.

By $p_b = 0$, we have that $f(0) = -r_b^k/d^k < 0$. In case $p_b = p_e$, $f(p_b) = \infty - r_b^k/d^k > 0$. Since $f(p_b)$ is continuous in the domain, there must be real solutions between 0 and p_e for (15).

Take the derivative of $f(p_b)$

$$\frac{df(p_b)}{p_b} = \frac{2p_b}{(p_e - p_b)^{(1/N_{CN}+1)}} + \frac{(1/N_{CN}+1)\cdot p_b^2}{(p_e - p_b)^{(1/N_{CN}+2)}}. \quad (A.1)$$

Note that $0 < p_b < p_e$, $N_{CN} \in Z$. As a result, $df(p_b)/dp_b > 0$. Therefore, there is only one real solution for $f(p_b) = 0$.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (61073186). Thanks are dur to the help of Xue Chen for her mathematical verification and Qiang Liu for his coding support.

References

[1] A. Liu, X. Jin, G. Cui, and Z. Chen, “Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network,” *Information Sciences*, vol. 230, pp. 197–226, 2013.

[2] S. Cui, A. J. Goldsmith, and A. Bahai, “Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1089–1098, 2004.

[3] M. Chen, M. Qiu, L. Liao, J. Park, and J. Ma, “Distributed multihop cooperative communication in dense wireless sensor networks,” *The Journal of Supercomputing*, vol. 56, no. 3, pp. 353–369, 2011.

[4] I. Ahmed, M. Peng, W. Wang, and S. I. Shah, “Joint rate and cooperative MIMO scheme optimization for uniform energy distribution in wireless sensor networks,” *Computer Communications*, vol. 32, no. 6, pp. 1072–1078, 2009.

[5] A. Liu, J. Ren, X. Li, Z. Chen, and X. S. Shen, “Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks,” *Computer Networks*, vol. 56, no. 7, pp. 1951–1967, 2012.

[6] A. F. Liu, X. Y. Wu, Z. G. Chen, and W. H. Gui, “Research on the energy hole problem based on unequal cluster-radius for wireless sensor networks,” *Computer Communications*, vol. 33, no. 3, pp. 302–321, 2010.

[7] J. Zhang, L. Fei, Q. Gao, and X.-H. Peng, “Energy-efficient multihop cooperative miso transmission with optimal hop distance in wireless ad hoc networks,” *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3426–3435, 2011.

[8] G. Yu, Z. Zhang, and P. Qiu, “Efficient ARQ protocols for exploiting cooperative relaying in wireless sensor networks,” *Computer Communications*, vol. 30, no. 14-15, pp. 2765–2773, 2007.

[9] S. S. Ikki and M. H. Ahmed, “Performance analysis of incremental-relaying cooperative-diversity networks over rayleigh fading channels,” *IET Communications*, vol. 5, no. 3, pp. 337–349, 2011.

[10] C. Zhang, W. Wang, and G. Wei, "Design of ARQ protocols for two-user cooperative diversity systems in wireless networks," *Computer Communications*, vol. 32, no. 6, pp. 1111–1117, 2009.

[11] H. Li, N. Jaggi, and B. Sikdar, "Relay scheduling for cooperative communications in sensor networks with energy harvesting," *IEEE Transactions on Wireless Communications*, vol. 10, no. 9, pp. 2918–2928, 2011.

[12] Z. Zhou, S. Zhou, J. H. Cui, and S. Cui, "Energy-efficient cooperative communication based on power control and selective single-relay in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 8, pp. 3066–3079, 2008.

[13] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, New York, NY, USA, 2005.

[14] A. Liu, Z. Liu, M. Nurudeen, X. Jin, and Z. 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.

[15] Z. Huang, H. Okada, M. Katayama, and T. Yamazato, "A study on cluster partitioning with cooperative miso scheme in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 490823, 9 pages, 2012.

[16] Y. Yuan, M. Chen, and T. Kwon, "A novel cluster-based cooperative MIMO scheme for multi-hop wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2006, Article ID 072493, 2006.

[17] K. Ota, M. Dong, Z. Cheng, J. Wang, X. Li, and X. S. Shen, "Oracle: mobility control in wireless sensor and actor networks," *Computer Communications*, vol. 35, no. 9, pp. 1029–1037, 2012.

[18] E. Larsson and P. Stoica, *Space-Time Block Coding for Wireless Communications*, Cambridge University Press, New York, NY, USA, 2008.

[19] Z. Zhou, S. Zhou, S. Cui, and J. H. Cui, "Energy-efficient cooperative communication in a clustered wireless sensor network," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3618–3628, 2008.

[20] S. K. Jayaweera, "Virtual MIMO-based cooperative communication for energy-constrained wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 5, pp. 984–989, 2006.

[21] A. Liu, D. Zhang, P. Zhang, G. Cui, and Z. Chen, "On mitigating hotspots to maximize network lifetime in multi-hop wireless sensor network with guaranteed transport delay and reliability," *Peer-to-Peer Networking and Applications*, pp. 1–19, 2012.

[22] L. Santaló, *Integral Geometry and Geometric Probability*, Cambridge University Press, New York, NY, USA, 2004.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 818917, 10 pages
http://dx.doi.org/10.1155/2013/818917

Research Article

Noncommutative Lightweight Signcryption for Wireless Sensor Networks

Lize Gu,[1] Yun Pan,[2] Mianxiong Dong,[3] and Kaoru Ota[4]

[1] *Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2] *School of Computer Science, Communication University of China, Beijing 100024, China*
[3] *School of Computer Science and Engineering, The University of Aizu, Aizu Wakamatsu 965-8580, Japan*
[4] *Department of Information and Electronic Engineering, Muroran Institute of Technology, Muroran 050-8585, Japan*

Correspondence should be addressed to Yun Pan; pany@cuc.edu.cn

Received 3 January 2013; Accepted 6 February 2013

Academic Editor: Anfeng Liu

Key management techniques for secure wireless-sensor-networks-based applications must minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility. Signcryption is the proper primitive to do this. However, existing signcryption schemes are heavyweight and not suitable for resource-limited sensors. In this paper, we at first propose a braid-based signcryption scheme and then develop a key establishment protocol for wireless sensor networks. From the complexity view, our proposal is 2^{15} times faster than RSA-based ones. As far as we know, our proposal is the first signcryption scheme based on noncommutative algebraic structures.

1. Introduction

Wireless sensor networks (WSNs) consist of a large number of micro, low-cost, low-power, and spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions [1, 2]. WSNs are often deployed in potentially adverse or even hostile environment so that there are concerns on security issues therein. To protect the confidentiality and privacy of WSN-oriented applications, the traditional symmetric (i.e., private-key), even lightweight, cryptography is often used. A well-known drawback to do this is that the symmetric cryptography is not as flexible as the asymmetric (i.e., public-key) cryptography. The main obstacle of using public-key cryptography in WSNs is that with limited memory, computing and communication capacity, and power supply, sensor nodes cannot employ sophisticated cryptographic operations such as modular exponentiation and pairing computation. Therefore, it is interesting to probe new efficient and lightweight implementations on some wellknown public-key cryptographic primitives, such as what has been done in TinyECC [3] and in MicroECC [4]. No matter which type cryptography is adopted, key establishment is one of the utmost concerns. At least, key establishment techniques for a secure WSN-based application must minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility [5].

Signcryption, now an international standard for data protection (ISO/IEC 29150, Dec 2011), was invented in 1996 and first disclosed to the public at CRYPTO 1997 [6, 7]. It is a data security technology by which confidentiality is protected and authenticity is achieved seamlessly at the same time. This will also allow smaller devices, such as smartphones and PDAs, 3G and 4G mobile communications, as well as emerging technologies, such as radio frequency identifiers (RFIDs) and wireless sensor networks, to perform high-level security functions. And, by performing these two functions simultaneously, we can save resources, be it an individual's time or be it energy, as it will take less time to perform the task. Therefore, signcryption is very suitable for key management in wireless sensor networks and other resource-constrained environments.

Since the invention of the primitive of signcryption, various constructions were proposed and most of them are based on three kinds of cryptographic assumptions. The first category assumes that the integer factoring problem (IFP) is intractable, such as the constructions in [8, 9]. The second category assumes that the discrete logarithm problem (DLP) over finite fields or elliptic curves (i.e., ECDLP) is intractable, such as the constructions in [10, 11]. In this category, some constructions further utilize the bilinear pairing to enhance the functionalities and performance, such as the constructions in [12, 13]. The third category is based on some lattice hard problems [14, 15]. Up to now, the last category attracts a lot of attention since the so-called quantum attack-resistant property. However, these existing lattice-based signcryptions have disadvantage in key sizes. Thus, it is interesting to probe new construction of signcryption based on other cryptographic primitives than IFP- and DLP-related ones and meanwhile keeping the potential of quantum attack resistance.

Under this background, some noncommutative groups have attracted the attention. One of the most popular groups in this category is the braid group. At CRYPTO 2000, Ko et al. [16] proposed the first fully fledged braid-based cryptosystem. In braid-based cryptographic schemes [16–24], the conjugacy search problem (CSP) (i.e., given two braids a and xax^{-1}, output the braid x) and its variants play a core role. Although many heuristic attacks, such as length-based attacks linear representation attacks, have obtained remarkable success in attacking braid-based cryptosystems and lowered the initial enthusiasm on this subject, there is no deterministic polynomial algorithms that can solve the CSP problem over braid groups [25] till now. On one hand, Birman et al. launched a project, referred to as BGGM project, to find polynomial algorithms for solving the CSP problem over Garside groups, including braid groups [26–28]. The BGGM project might be the strongest efforts known for solving the CSP problem over braid groups in polynomial-time (with respect to the input size). Up to now, the BGGM project has already made a great progress; except for rigid pseudo-Anosolov braids, the CSP instances over other braids can be solved in polynomial time [28]. On the other hand, some researchers still keep on finding hard instances of the CSP problem in braid groups. For examples, in 2007, Ko et al. [29] proposed some ideas on generating hard instances for braid cryptography, and in 2010, Prasolov [30] constructed some small braids with large ultra summit set (USS). Prasolov's result represents a frustration toward the BGGM project, but an encouragement toward the intractability assumption of the CSP problem over braid groups. According to [31], if p and s are random braids, then the length of sps^{-1} is, with a high probability, about the length of p plus the double of the length of s. This is the reason why the length-based attacks work. This also suggests that one can defeat the length-based attacks by requiring that the length of sps^{-1} is closer to the length of p. This in turn requires that p should lie in its super summit set (SSS) [31]. We know that USS ⊂ SSS. Therefore, if we can work with the braids suggested by Prasolov, then we reach the point to instantiate our proposal with braid groups in a secure manner.

Another promising observation coming from [23] is that braid operations can be implemented with a complexity level of about 2^{15} bit operations, while the complexity level of the exponentiation over 1024 bit RSA modular is about 2^{30} bit operations. This suggests that braid-based cryptosystems admit ultra efficient, even lightweight, implementations.

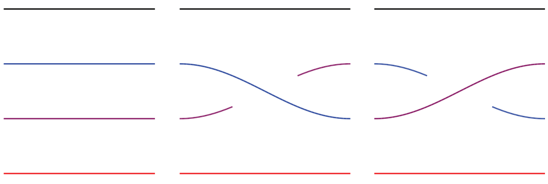

(a) e (b) σ_2 (c) σ_2^{-1}

Figure 1: Geometrical illustration on identity and Artins generators [23].

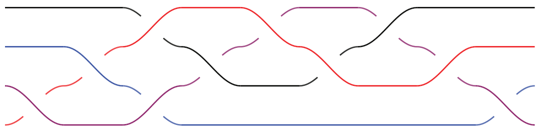


Figure 2: An example of geometric braids [23].

The main motivation of this paper covers two aspects: the first is to design a lightweight signcryption scheme based on noncommutative groups assuming that the CSP problem over the underlying groups are intractable, and the second is to construct efficient key management protocols for wireless sensor networks.

The rest contents are organized as follows. In Section 2, we at first give a simple introduction to the braid group, and then introduce the left self-distributive system and its properties. A building block—braid-based signcryption scheme is proposed in Section 3, and the full description of the key management protocol for wireless sensor networks is developed in Section 4. Performance evaluation and comparisons, including security level analysis, are given in Section 5, respectively. Concluding remarks are given in Section 6.

2. Preliminaries

2.1. Braid Group and Related Cryptographic Problems. The n-braid group B_n is presented by the Artin generators $\sigma_1, \ldots, \sigma_{n-1}$ and relations $\sigma_i \sigma_j = \sigma_j \sigma_i$ for $|i - j| > 1$ and $\sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_j$ for $|i - j| = 1$ $(1 \leq i, j \leq n - 1)$. Braid groups also admit a very intuitively geometrical illustration: the identity of braid groups, that is, the empty braid e, and the Artin generators (e.g., $\sigma_2^{\pm 1}$ in B_4) as shown in Figure 1 [23].

Geometrically, the product of two braids is the braid obtained by merging the tail of the first braid with the head of the second braid. For example, Figure 2 shows the braid $\sigma_1 \sigma_2 \sigma_1^{-1} \sigma_3^{-1} \sigma_2 \sigma_3 \sigma_2 \sigma_3^{-1} \sigma_2^{-1} \sigma_1$ [23].

There is a natural automorphism from B_2 to the integer additive group $\mathbb{Z}$ and this means that B_2 is infinite and commutative. But for $n \geq 3$, the braid group B_n is infinite and noncommutative. In addition, for each m $(\leq n)$, the identity mapping on $\{\sigma_1, \ldots, \sigma_{m-1}\}$ naturally induces an embedding of B_m into B_n [23].

For arbitrary two braids $x, y \in B_n$, we say they are conjugate, written as $x \sim y$, if $y = a^{-1}xa$ for some $a \in B_n$. Here a or a^{-1} is called a conjugator. The conjugacy deciding problem (CDP) is to determine whether $x \sim y$ for a given instance $(x, y) \in B_n^2$, while the conjugator searching problem (CSP) is to find a braid $z \in B_n$ such that $y = z^{-1}xz$ for a given instance $(x, y) \in B_n^2$ with $x \sim y$. At present, we know that both CDP and CSP over braid groups are solvable; that is, there is a deterministic algorithm that stops after finite steps, not necessarily polynomially bounded, and outputs an accurate solution. However, it seems that both of them are, at least in worst cases, intractable; that is, there is no probabilistic polynomial time algorithms that output an accurate solution with nonnegligible probability (with respect to the length of description of the input instances) [20, 21, 23].

In sequel, we use x^a to denote the conjugate braid $a^{-1}xa$ when $a \in B_n$. Meanwhile, we also use x^t to denote the multiplication braid $\underbrace{x \cdots x}_{t \text{ times}}$ when $t \in \mathbb{N}$.

2.2. Conjugacy-Based Left Self-Distributive Systems. Under the intractability assumption of the conjugator search problems over certain noncommutative semigroups, Wang et al. [24] proposed several public-key cryptosystems based on conjugacy-based left self-distributive systems. The notations and related constructions are helpful for developing our main proposal in this paper. Therefore, let us recall the definition of the left self-distributive system that was firstly postulated by Dehornoy [32].

Definition 1 (left self-distributive system LD [32]). Suppose that S is a nonempty set, $F : S \times S \to S$ is a well-defined function and let us denote $F(a, b)$ by $F_a(b)$. If the following rewritten formula holds

$$F_r(F_s(p)) = F_{F_r(s)}(F_r(p)), \quad (\forall p, r, s \in S), \tag{1}$$

then, we call $F_\cdot(\cdot)$ a left self-distributive system, abbreviated as LD system.

The terminology "left self-distributive" arises from the following analogical observation: if we consider $F_r(s)$ as a binary operation $r * s$, then the formula (1) becomes

$$r * (s * p) = (r * s) * (r * p); \tag{2}$$

that is, the operation "$*$" is left self-distributive with respect to itself [32].

One can define the following LD system, named as Conj-LD system, which means an abbreviation of left self-distributive system defined by conjugate operations.

Definition 2 (Conj-LD system [24]). Let G be a noncommutative semigroup and $G^{-1} \subset G$ the set of all invertible elements. The binary function F given by the following conjugate operation:

$$F : G^{-1} \times G \longrightarrow G, \qquad (a, b) \longmapsto a^{-1}ba \triangleq b^a \tag{3}$$

is an LD system, abbreviated as Conj-LD.

It is easy to see that F caters to the rewritten formula (1). Thus, $F_a(b)$ is an LD system [24].

Table 1: Experiments for define CSP-DDH problem.

Experiment $\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-real}}$	Experiment $\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-rand}}$
$i \xleftarrow{\$} \mathbb{N}; X \leftarrow F_{a^i}(b);$	$i \xleftarrow{\$} \mathbb{N}; X \leftarrow F_{a^i}(b);$
$j \xleftarrow{\$} \mathbb{N}; Y \leftarrow F_{a^j}(b);$	$j \xleftarrow{\$} \mathbb{N}; Y \leftarrow F_{a^j}(b);$
$Z \leftarrow F_{a^{i+j}}(b);$	$\ell \xleftarrow{\$} \mathbb{N}; Z \leftarrow F_{a^\ell}(b);$
$b \leftarrow \mathscr{A}(X, Y, Z);$	$b \leftarrow \mathscr{A}(X, Y, Z);$
Return b.	Return b.

Proposition 3 (power law [24]). *Let F be a Conj-LD system defined over a noncommutative semigroup G. Suppose that $a \in G^{-1} \subset G$ and $b \in G$ are given and fixed. Then, for arbitrary three positive integers m, s, and t such that $m = s + t$, one has*

$$\begin{gathered} F_a(b^m) = F_a(b^s) F_a(b^t) = F_a^m(b), \\ F_{a^m}(b) = F_{a^s}(F_{a^t}(b)). \end{gathered} \tag{4}$$

Remark 4. By using the notation of $F_\cdot(\cdot)$, the intractability assumption of the CSP problem in G can be reformulated as follows: it is hard to retrieve a' from the given pair $(a, F_a(b))$ such that $F_a(b) = F_{a'}(b)$ (see more details in [24]).

Definition 5 (CSP-based decisional Diffie-Hellman: CSP-DDH [24]). Let F be a Conj-LD system defined over a noncommutative semigroup G and let $\mathscr{A}$ be an adversary. For arbitrary $a \in G^{-1}$ and $b \in G$, consider the following two experiments in a paralleled manner (see Table 1). Now define the advantage of $\mathscr{A}$ in violating the CSP-based decisional Diffie-Hellman assumption as

$$\begin{aligned} \mathbf{Adv}_{F,\mathscr{A}}^{\text{csp-ddh}} = \Big| & \Pr\left[\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-real}} = 1\right] \\ & - \Pr\left[\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-rand}} = 1\right] \Big|. \end{aligned} \tag{5}$$

Intuitively, the CSP-DDH assumption states that the distributions:

$$\begin{gathered} \mathscr{D}_1 \triangleq \left(F_{a^i}(b), F_{a^j}(b), F_{a^{i+j}}(b)\right), \\ \mathscr{D}_2 \triangleq \left(F_{a^i}(b), F_{a^j}(b), F_{a^\ell}(b)\right) \end{gathered} \tag{6}$$

are computationally indistinguishable when $i, j, \ell \in \mathbb{N}$ are drawn at random.

Remark 6. Intuitively, it is hard to solve the CSP-DDH problem without solving the CSP problem if G is modeled as a generic semigroup model. According to [33], we know that the discrete logarithm problem (DLP) over finite fields and the corresponding DDH problem are polynomially equivalent in a generic cyclic group. By an analogical manner, we speculate that the CSP problem and the CSP-DDH problem in a generic noncommutative semigroup are polynomially equivalent (see more details in [24]).

2.3. The Fujisaki-Okamoto Transformation [34, 35]. Without loss of generality, a public-key encryption scheme can be defined as a triple $\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D})$, where

(i) $\mathscr{K}$ is the key generation algorithm that takes as input a system security parameter 1^k and outputs a public-/private-key pair (pk, sk). In general, this algorithm can be formulated as $(pk, sk) \leftarrow \mathscr{K}(1^k)$.

(ii) $\mathscr{E}$ is the encryption algorithm that takes as inputs the public-key pk and a message $m \in \mathscr{M}$ and outputs a ciphertext $c \in \mathscr{C}$, where $\mathscr{M}$ and $\mathscr{C}$ are message space and ciphertext space, respectively. In general, this algorithm can be formulated as $c \leftarrow \mathscr{E}_{pk}(m)$ or $c \leftarrow \mathscr{E}_{pk}(m; r)$ when it is necessary to specify the random salt r used in the encryption process.

(iii) $\mathscr{D}$ is the decryption algorithm that takes as inputs the secret key sk and a ciphertext $c \in \mathscr{C}$ and outputs a message $m \in \mathscr{M}$ or a symbol $\perp$, which indicates that c is invalid. In general, this algorithm can be formulated as $m/ \perp \leftarrow \mathscr{D}_{sk}(c)$.

In general, as for public-key encryption, one-wayness against chosen plaintext attacks (OW-CPA) is the lowest security requirement, while indistinguishability against adaptively chosen ciphertext attacks (IND-CCA2) is the most desirable and the standard security requirement. Cryptographic practise shows that it is always easier to design an OW-CPA secure encryption scheme than to directly design an IND-CCA2 secure one. Thus, it is desirable to have a general method for transforming an OW-CPA secure encryption scheme to an IND-CCA2 secure one [35]. Fortunately, one of this methods was invented by Fujisaki and Okamoto [34] at PKC 1999.

Theorem 7 (FO transformation [34]). *Suppose H_1 and H_2 are two random oracles with required domains and ranges, respectively. Given a public-key encryption scheme*

$$\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D}) \tag{7}$$

that achieves the security of one-wayness against chosen plaintext attacks (OW-CPA), one can get another public-key encryption scheme

$$\pi' = \left(\mathscr{K}', \mathscr{E}', \mathscr{D}'\right) \tag{8}$$

that achieves the security of indistinguishability against adaptively chosen ciphertext attacks (IND-CCA2), where

(1) *key generation algorithm $\mathscr{K}'$ is identical to $\mathscr{K}$;*

(2) *encryption algorithm is defined as*

$$\mathscr{E}'_{pk}(m) = \left(\mathscr{E}_{pk}(r), m \oplus H_1(r), H_2(m, r)\right), \tag{9}$$

where r is picked at random;

(3) *decryption algorithm $\mathscr{D}'_{sk}(c_1, c_2, c_3)$ performs the following steps:*

(a) $r' \leftarrow \mathscr{D}_{sk}(c_1)$;

(b) $m' \leftarrow c_2 \oplus H_1(r')$;

(c) *output m' if $c_3 = H_2(m', r')$ and $\perp$ otherwise.*

3. Building Block: Noncommutative Signcryption

Before describing our proposal for WSN key management, let us at first propose a signcryption scheme from noncommutative semigroups where the CSP-related assumptions hold. We will see later, when this scheme is instantiated by using braids, we obtain a very efficient signcryption scheme that is 2^{15} times faster than RSA-based signcryption (suppose that 1024 bit RSA modulus were used).

Suppose that G is a noncommutative semigroup so that the CSP problem and the CSP-DDH problem over G are intractable. Then, the public parameters of the proposed signcryption are given by a quintuple $\langle \mathfrak{D}, a, b, H_1, H_2 \rangle$, where

(i) $\mathfrak{D}$ is a description of G and $G^{-1} \subset G$. Without loss of generality, we assume the length of $\mathfrak{D}$ is bounded by $\mathscr{O}(\log |G|)$ for finite G. When G is infinite but admits a finite presentation, say $fp(G) = \langle X \mid R \rangle$, the length of $\mathfrak{D}$ is the sum of the length of X and the length of R. However, for braid group B_n, $\mathfrak{D}$ admits even efficient description since whenever the braid index n is given, the generator set $X = \{\sigma_1, \ldots, \sigma_{n-1}\}$ and the relation set $R = \{\sigma_i \sigma_j = \sigma_j \sigma_i : |i - j| > 1\} \cup \{\sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_j : |i - j| = 1\}$ $(1 \leq i, j \leq n - 1)$ is totally specified. That is, for braid group B_n, $\mathfrak{D} = n$;

(ii) $a \in G^{-1} \subset G$ and $b \in G$ are two fixed elements that are picked at random;

(iii) $H_1 : G \rightarrow G^2$ and $H_2 : G^2 \rightarrow G$ are two cryptographic hash functions that are modeled as random oracles.

Then, the proposed signcryption scheme consists of the following three algorithms:

(i) $\mathscr{K}\mathscr{G}(1^k)$, key generation algorithm that takes as input the system security parameter 1^k, picks an integer $s \in \{0,1\}^k$ at random calculates $x = b^{a^s} \in G$, and finally outputs (s, x) as the private-/public-key pair.

(ii) $\mathscr{S}\mathscr{C}(s, y; m)$, signcryption algorithm that takes as inputs the sender's private-key $s \in \{0,1\}^k$, the receiver's public-key $y \in G$, and the message $m \in G$, and performs the following steps:

(1) pick $t \in \{0,1\}^k$ at random;

(2) compute

$$\begin{gathered} c_1 = b^{a^t}, \\ h = H_2(m, c_1), \\ \sigma = \left(a^t\right)^{-1} a^s h c_1, \\ c_2 = (m \parallel \sigma) \oplus H_1\left(y^{a^t}\right), \end{gathered} \tag{10}$$

where operator "$\oplus$" should be viewed as XOR operation over bit-strings that are encoding results of a pair in G^2;

(3) output (c_1, c_2).

Theorem 8. *The proposed signcryption is consistent.*

Proof. Suppose that the sender and the receiver performs honestly, and their inputs are well formed. That is, $x = b^{a^s}$ and $y = b^{a^r}$. Then, since

$$\begin{aligned}
c_1^{a^r} &= \left(b^{a^t}\right)^{a^r} \\
&= b^{a^{t+r}} \\
&= \left(b^{a^r}\right)^{a^t} \\
&= y^{a^t}, \\
m' \parallel \sigma' &= c_2 \oplus H_1\left(c_1^{a^r}\right) \\
&= (m \parallel \sigma) \oplus H_1\left(y^{a^t}\right) \oplus H_1\left(y^{a^t}\right) \\
&= m \parallel \sigma, \\
h' &= H_2\left(m', c_1\right) \\
&= H_2\left(m, c_1\right) \\
&= h,
\end{aligned} \tag{11}$$

we have

$$\begin{aligned}
c_1^{\sigma'} &= \left(b^{a^t}\right)^{\sigma} \\
&= \left(b^{a^t}\right)^{(a^t)^{-1} a^s h c_1} \\
&= \left(b^{a^s}\right)^{h c_1} \\
&= x^{h c_1} \\
&= x^{h' c_1}.
\end{aligned} \tag{12}$$

Then, $m' = m$ will be output correctly. □

Theorem 9. *Suppose that H_1 and H_2 are random oracles. The proposed signcryption is indistinguishable against adaptively chosen ciphertext attack (IND-CCA2) assuming that the CSP-DDH problem over the underlying noncommutative semigroup G is intractable.*

Proof. To apply the well-known Fujisaki-Okamoto transformation theorem [34], we at first need to define an IND-CPA secure encryption scheme $\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D})$ and then establish the security relationship between the proposed signcryption scheme and the enhanced encryption scheme π', that is, an FO transformation from π. This can be done by setting $\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D})$ as follows:

(i) $\mathscr{K}(1^k) := \mathscr{KG}(1^k)$. That is, the key generation algorithm remains unchanged.

(ii) The encryption algorithm $\mathscr{E}(y; m)$ that takes as inputs the receiver's public-key $y \in G$ and the intended message $m \in G$ and then performs the following steps:

  (1) pick $t \in \{0,1\}^k$ at random;
  (2) compute $c_1 = b^{a^t}$ and $c_2 = y^{a^t} m$;
  (3) output (c_1, c_2).

(iii) The decryption algorithm $\mathscr{D}(r; c_1, c_2)$ that takes as inputs the receiver's private-key $r \in \{0,1\}^k$ and the ciphertext pair $(c_1, c_2) \in G^2$ and then outputs the intended message $m = c_2(c_1^{a^r})^{-1}$.

Apparently, this is just the ElGamal-like variant based on CSP-DDH assumption. According to Theorem 1 of [24], this is IND-CPA secure. Then, according to Theorem 7, the FO variant $\pi' = (\mathscr{K}', \mathscr{E}', \mathscr{D}')$ is IND-CCA2 secure when H_1 and H_2 are modeled as random oracles, where

(i) $\mathscr{K}'(1^k) := \mathscr{K}(1^k)$.

(ii) $\mathscr{E}'(y; m)$ performs the following steps:

  (1) pick $u \in G$ at random;
  (2) let $(c_1, c_2) \leftarrow \mathscr{E}(y; u)$;
  (3) let $c_3 = m \oplus H_1(u)$ and $c_4 = H_2(m, u)$;
  (4) output (c_1, c_2, c_3, c_4).

(iii) The decryption algorithm $\mathscr{D}(r; c_1, c_2, c_3, c_4)$ that takes as inputs the receiver's private-key $r \in \{0,1\}^k$ and the ciphertext qudruple (c_1, c_2, c_3, c_4), and then performs the following steps:

  (1) let $u' \leftarrow \mathscr{D}(r; c_1, c_2)$;
  (2) let $m' \leftarrow c_3 \oplus H_1(u')$;
  (3) output m' if $c_4 = H_2(m', u')$ and $\perp$ otherwise.

Now, let us show that in the same random oracle models, if there is a polynomlai-time adversary $\mathscr{A}$ that can, with non-negligible probability, break the IND-CCA2 security of the proposed signcryption scheme, there is another polynomial-time adversary $\mathscr{B}$ that can, by controlling the response of the random oracles H_1 and H_2, break the IND-CCA2 security of π'. However, this is contrary to the fact that π' is IND-CCA2 secure. Therefore, $\mathscr{A}$'s advantage of breaking the proposed signcryption scheme must be negligible.

In fact, if $\mathscr{B}$ controls the response of the random oracles H_1 and H_2, then it can break the IND-CCA2 security of π' with nonnegligible probability. This is apparently, since B controls the response of H_2, whenever seeing a ciphertext (c_1, c_2, c_3, c_4), it can retrieve the message m and random salt u by looking up the response list of H_2 under the reasonable assumption that the probability for different pair (m', u') with same hash value with the pair (m, u) is negligible.

The left thing is to show that $\mathscr{B}$, without knowing the receiver's private-key $r \in \{0,1\}^k$, how to simulate the response on decryption queries for $\mathscr{A}$ in a perfect manner. Whenever

A invokes a decryption query by submitting a signcryption pair (c_1, c_2), $\mathscr{B}$ responds as follows:

(1) look up (h_2, m_i, c_1) in H_2-list. If there is no matched triple, $\mathscr{B}$ sends $\perp$ to $\mathscr{A}$ as the response;

(2) for each matched triple (h_2, m_i, c_1), $\mathscr{B}$ performs the following steps:

(a) for each (h_1, Y_i) in H_1-list, do the following steps:

(i) extract a possible σ_i according to the following formula:

$$c_2 = (m_i \parallel \sigma_i) \oplus h_1. \tag{13}$$

This can be done since $\mathscr{B}$ knows c_2, m_i and h_1 at this stage;

(ii) test whether the equality $c_1^{\sigma_i} = x^{h_2 c_1}$ holds? (recall that x is the verification key of the singer). If so, replies $\mathscr{A}$ with m_i and end of the response; otherwise, continue;

(3) if up to now, $\mathscr{B}$ has not output response to $\mathscr{A}$ yet, then $\mathscr{B}$ sends $\perp$ to $\mathscr{A}$ as the response.

Now, let us show that $\mathscr{B}$'s simulation is perfect. It is reasonable to assume that without accessing hash queries on H_1 and H_2, $\mathscr{A}$'s probability for submitting a valid signcryption pair (c_1, c_2) is negligible. Thus, whenever $\mathscr{A}$ invokes hash queries on H_1 and H_2 for forming a valid signcryption pair, related materials are recorded and $\mathscr{B}$ can retrieve them and finally send $\mathscr{A}$ a perfect response. □

Remark 10. Note that although the signature scheme embedded in the proposed signcryption scheme merely achieves unforgeable against no-message attacks, the resulted signcryption is existentially unforgeable against external adaptively chosen message attack. Here, external forgeries means that it is neither the singer, nor the intended receiver. We know that it is reasonable to exclude the signer from forgeries. Let us explain why we further exclude the intended receiver from the forgeries. In fact, the primitive of signcryption provides confidentiality of the message against all entities except the intended receiver and meanwhile it provides the authenticity of the sender (i.e., the signer) for the intended receiver. That is, the authenticity embedded in the signcryption primitive is unidirectional, instead of bidirectional. Therefore, it seems that there is no reason for an intended receiver to forge a signature on behalf of some signer and then encrypt the signature for himself/herself, except for planting false evidence against some senders. In other words, in our proposal, we assume that the receiver who possesses the corresponding private-key for performing designcryption is honest. Otherwise, an existentially unforgeable signature scheme, such as the noncommutative signature scheme in [36] should be embedded therein. For further consideration of the insider security and the outsider security of signcryptions, one can refer to [37, 38].

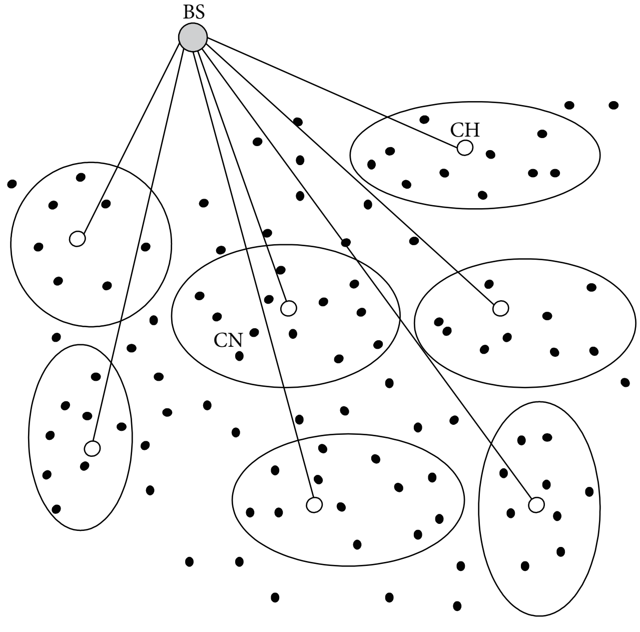


Figure 3: WSN Architecture.

4. Lightweight Implementation of Key Management Protocols for WSNs

In [5], Hagras et al. described an efficient key management scheme for WSNs based on elliptic curve signcryption. Our proposal follows their diagram. However, the main differences of our work lie in the following aspects:

(i) firstly, the signcryption algorithm used by Hagras et al. is abstract and essentially hybrid where a symmetric encryption algorithm is involved. However, we will give a detailed specification of each algorithm;

(ii) secondly, Hagras et al.'s proposal is based on commutative platforms, while as far as we known, our proposal is firstly based on noncommutative platforms.

Similar to [5], suppose that the network architecture is the standard clustered WSN architecture depicted in Figure 3. The proposed key management scheme supports three protocols: the first is used to generate private-/public-keys for each individual nodes, including base nodes, cluster headers, and cluster nodes; the second is essentially a signcryption scheme that is used by base node to send session keys to cluster heads; and the third is essential also a signcryption scheme that is used by cluster heads to send session keys to cluster nodes.

Let B_n be the braid group and $a, b \in B_n$. Suppose that $F(\cdot,\cdot)$ is the Conj-LD system defined over braid group B_n, while $H_1 : G \to G^2$ and $H_2 : G^2 \to G$ are two cryptographic hash functions. Our proposal consists of three protocols that are described in the following subsections.

4.1. Key Generation Protocol. This protocol is responsible for creating public-/private-key pairs for base nodes (BNs), cluster heads (CHs), and cluster nodes (CNs).

Step 1. Generate public-/private-key for based nodes.

$V_{\mathrm{BN}} \in \{0,1\}^k$: the private-key for the base node is a positive integer chosen uniformly at random.

$P_{\mathrm{BN}} \in B_n$: the corresponding public-key for the base node is calculated as $P_{\mathrm{BN}} = F(a^{V_{\mathrm{BN}}}, b)$.

Step 2. Generate public-/private-key for cluster heads.

$V_{\mathrm{CH}_j} \in \{0,1\}^k$: the private-key for the jth cluster head is a positive integer chosen uniformly at random.

$P_{\mathrm{CH}_j} \in B_n$: the corresponding public-key for the jth cluster head is calculated as $P_{\mathrm{CH}_j} = F(a^{V_{\mathrm{CH}j}}, b)$.

Step 3. Generate public-/private-key for cluster nodes.

$V_{\mathrm{CN}_i} \in \{0,1\}^k$: the private-key for the ith cluster head is a positive integer chosen uniformly at random.

$P_{\mathrm{CN}_i} \in B_n$: the corresponding public-key for the ith cluster head is calculated as $P_{\mathrm{CN}_i} = F(a^{V_{\mathrm{CN}i}}, b)$.

Step 4. Session key generation for base node and cluster heads.

(1) The base node creates the session key $K_{\mathrm{BN-CH}_j}$ which will be used for secure communication between the jth cluster head and the base node.

(2) The jth cluster head creates the session key $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ which will be used for secure communication between the jth cluster head and the ith cluster node.

Without loss of generality, here we assume that $K_{\mathrm{BN-CH}_j}$ and $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ are elements of G picked at random. (In fact, we can always employ an encoding algorithm to map elements of G into valid session keys.)

Remark 11. Note that in the last step, all session keys are newly generated by the base node and the cluster nodes, respectively. In fact, after the execution of Steps 1, 2 and 3, we know that the base node and the jth cluster head can calculate the shared session key $K_{\mathrm{BN-CH}_j} = F(a^{V_{\mathrm{BN}}+V_{\mathrm{CH}j}}, b)$, and the jth cluster head and the ith cluster node can calculate the shared session key $K_{\mathrm{CH}_j-\mathrm{CN}_i} = F(a^{V_{\mathrm{CH}j}+V_{\mathrm{CN}i}}, b)$. However, it is not a good choice to use this kind of session keys since they are totally determined by long-term private-keys. Instead, we suggest to renew a session key instantly to guarantee its freshness.

4.2. BN-CHs *Signcryption.* The base node signcrypts the session key $K_{\mathrm{BN-CH}_j}$ using its private-key and sends the ciphertext (c_1, c_2) to the jth cluster head as follows:

(1) pick $t \in \{0,1\}^k$ at random;

(2) $c_1 = F(a^t, b)$;

(3) $h = H_2(K_{\mathrm{BN-CH}_j}, c_1)$;

(4) $\sigma = (a^t)^{-1} a^{V_{\mathrm{BN}}} h c_1$;

(5) $c_2 = (K_{\mathrm{BN-CH}_j} \| \sigma) \oplus H_1(F(a^t, P_{\mathrm{CH}_j}))$;

(6) send (c_1, c_2) to the jth cluster head.

Upon receiving the ciphertext (c_1, c_2) from the base node, the jth cluster head designcrypts the session key as follows:

(1) compute $K \| \sigma = c_2 \oplus H_1(F(a^{V_{\mathrm{CH}j}}, c_1))$, $h = H_2(K, c_1)$;

(2) accept K if $F(\sigma, c_1) = F(hc_1, P_{\mathrm{BN}})$ and report "FAILURE" otherwise.

4.3. CH-CNs *Signcryption.* The jth cluster head signcrypts the session key $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ using its private-key and sends the ciphertext (d_1, d_2) to the ith cluster node as follows:

(1) pick $s \in \{0,1\}^k$ at random.

(2) $d_1 = F(a^s, b)$.

(3) $g = H_2(K_{\mathrm{CH}_j-\mathrm{CN}_i}, d_1)$.

(4) $\sigma = (a^s)^{-1} a^{V_{\mathrm{CH}j}} g d_1$.

(5) $d_2 = (K_{\mathrm{CH}_j-\mathrm{CN}_i} \| \sigma) \oplus H_1(F(a^s, P_{\mathrm{CN}_i}))$.

(6) Send (d_1, d_2) to the ith cluster node.

Upon receiving the ciphertext (d_1, d_2) from the jth cluster head, the ith cluster node designcrypts the session key as follows:

(1) compute $K \| \sigma = d_2 \oplus H_1(F(a^{V_{\mathrm{CN}i}}, d_1))$, $h = H_2(K, d_1)$;

(2) accept K if $F(\sigma, d_1) = F(gd_1, P_{\mathrm{CH}_j})$ and report "FAILURE" otherwise.

5. Performance Evaluation

5.1. Complexity of Basic Operations. Now, let us compare the braid-based signcryption schemes with the RSA-based ones. According to Cha et al.'s implementation [39] and Maffre's test [40], the complexities of the braid operations, such as multiplication, inversion, and canonical form computation, are bounded by $\mathcal{O}(l^2 n \log n)$ in the sense of bit operations, where n and l are the braid index and the canonical length of involved braids, respectively. If we follow Maffre's suggestions by setting $n = 50$ and $l = 10$, then the number of bit operations for implementing these braid operations is proportional to 2^{15}. We know that the number of bit operations for implementing modular exponentials involved in RSA-based schemes is proportional to 2^{30} when the bit length of RSA modulus is set to 1024. This suggests that the proposed braid-based signcryption is about 2^{15} times faster than RSA-based ones.

Further, if we lift the security level of the RSA-based schemes to exp(92.80), which is comparable to the security level of our scheme (see Section 5.3), then the RSA modulus should be at least 2008 bits (see [23] for details). Then, the number of bit operations for implementing modular exponentials involved in RSA-based schemes is proportional to 2^{33}. This suggests that at the same security level, our braid-based signcryption is even efficient than that of RSA-based ones.

Table 2: Parameter length.

Parameter	Components and domains	Size	Size in bits ($n = 50, l = 10$)
System parameters	$n \in \mathbb{N}, a, b \in B_n$	$\lceil \log n + 2 \ln \log n \rceil$	5650
Private key[1]	$s \in \{0, 1\}^k$	k	80
Public key	$b^{as} \in B_n$	$\lceil \ln \log n \rceil$	2822
Signcryption[2]	$(c_1, c_2) \in B_n \times B_n^2$	$\lceil 3 \ln \log n \rceil$	8466
Total	—	$\approx \lceil 6 \ln \log n \rceil$	≈17 K

[1] It is enough to use 80-bit private keys in WSN-oriented applications.

[2] The length of c_2 is about equivalent to the length of two braids.

Table 3: Complexities and security levels.

	RSA-based schemes [23]			Braid-based schemes	
	Technique	$k = 1024$	$k = 2008$	Technique	$n = 50, l = 10$
Signcryption	Modular Exp.	2^{30}	2^{33}	Braid operation	2^{15}
Security level[1]	Factoring	exp (69.69)	exp (92.80)	Solving CSP	exp (92.80)

[1] The security level of RSA-based schemes are evaluated according to the best known factoring method, that is, the number field sieve (NFS) method [41].

5.2. Parameter Size. A braid in B_n with l canonical factors can be represented by a bit string of size $\lceil \ln \log n \rceil$ [16]. Thus, when $n = 50$ and $l = 10$, the sizes of the system parameters, the private-key, the public-key, and the ciphertexts are 5650 bits, 80 bits, 2822 bits, and 8466 bits, respectively. In total, it is about 17 Kbits (see Table 2). According to [5], a typical WSN node, MICA2 mote, developed by the University of California at Berkeley has an 8-bit 7.3 MHz processor with 4 KB (i.e., 32 Kbits) RAM and 128 KB programmable ROM. This suggests that although our scheme will take more memory than RSA-based ones, it is still compact enough to be deployed in typical WSN environments.

5.3. Security Levels. In [23], Wang et al. presented an analysis of the security levels of braid-based cryptosystems against two typical attacks: heuristic attacks and brute force attacks. In a similar manner, we can discuss the security levels of the proposed signcryption scheme. According to [23], the security level of a cryptosystem is modeled as the number of bit operations for breaking the cryptosystem. Since this number is in general huge, we always use its logarithm in evaluation and refer to as the logarithmic security level.

As for braid-based cryptosystems, heuristic attacks mean currently known smart attacks, such as length-based attacks [42, 43] and linear representation attacks. According to Maffre's test [40] and Wang et al.'s summarization [23], the logarithmic complexity of existing heuristic attacks against braid-based cryptosystems can be expressed as $\log(C_{150}^{50}) \approx 92.80$.

Let us proceed to analyze the security level against brute force attacks. According to Ko et al. [29], when the private-keys of braid-based schemes are selected carefully, that is, avoiding the weak keys mentioned by Maffre [40], all known heuristic attacks will be unsuccessful. Further, according to the previous analysis given by Ko et al. [16], the complexity of carrying brute force attacks towards braid-based schemes is proportional to $\exp((1/2) \ln \log n)$. Therefore, when we adopt Maffre's suggestion by setting the braid index and the canonical length of the involved braids to $n = 50$ and $l = 10$, respectively, the security level of our scheme against brute force attacks is proportional to exp(978). This suggests that in the foreseeable future it is infeasible to launch exhaustive attacks towards our proposal.

In brief, we can summarize the performance comparisons in two cases: in Case I, we consider the currently acceptable parameter settings, and in Case II, we lift the security level of the RSA-based schemes to exp(92.80) by increasing the length of the corresponding RSA modulus. The results are listed in Table 3. We can conclude that our scheme is very fast in signcrypting and designcrypting, but acceptably larger in storage requirement.

Remark 12. Although Table 3 seems very similar to that in [23], there are remarkable differences as follows: on one hand, in [23], the efficiencies of the signing process and the verifying process of the braid-based signature scheme in [23] are much different; signing can be implemented in the complexity proportional to 2^{15}, while the complexity of verifying is proportional to 2^{34}. However, the efficiencies of the signcrypting process and the designcrypting process in this paper are same: both of them are proportional to 2^{15} since in our new proposal it is unnecessary to solve the CDP problem over braid groups; on the other hand, the braid-based scheme in [23] is merely a signature scheme, while the proposal in this paper is a signcryption scheme. This suggests that our signcryption scheme is much efficient than Wang et al.'s signature scheme [23]. In brief, our proposal does more and faster than that in [23].

6. Conclusion

Lightweight cryptographic schemes are useful for securing WSN-oriented applications. To minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility, signcryption is the proper primitive to realize key management protocols for WSNs. However, most existing signcryption schemes are heavyweight and not suitable for resource-limited sensors. In this paper, we propose a braid-based signcryption scheme and then develop a key establishment protocol for wireless sensor networks. From the complexity view, the proposed scheme is 2^{15} times faster than RSA-based ones. As far as we know, this proposal is the first signcryption scheme based on noncommutative algebraic structures. In addition, the analysis of the basic operations and parameter sizes suggests that our proposal can be efficiently deployed in typical WSN environments.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC) (nos. 61003285, 61070251, 61103198), the NSFC A3 Foresight Program (no. 61161140320) and the JSPS A3 Foresight Program, JSPS Research Fellowships for Young Scientists Program, and NEC C&C Foundation.

References

[1] M. Dong, K. Ota, X. Li, X. Shen, S. Guo, and M. Guo, "HARVEST: a task-objective efficient data collection scheme in wireless sensor and actor networks," in *Proceedings of the 3rd International Conference on Communications and Mobile Computing (CMC '11)*, pp. 485–488, April 2011.

[2] K. Ota, M. Dong, and X. Li, "TinyBee: mobile-agent-based data gathering system in wireless sensor networks," in *Proceedings of the IEEE International Conference on Networking, Architecture, and Storage (NAS '09)*, pp. 24–31, IEEE Press, July 2009.

[3] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 245–256, April 2008.

[4] M. Varchola, T. Guneysu, and O. Mischke, "MicroECC: a lightweight reconfigurable elliptic curve crypto-processor," in *Proceedings of the International Conference on Reconfigurable Computing and FPGAs (RECONFIG '11)*, pp. 204–210, IEEE Computer Society, Washington, DC, USA, 2011.

[5] E. A. A. A. Hagras, D. El-Saied, and H. H. Aly, "Energy efficient key management scheme based on elliptic curve signcryption for Wireless Sensor Networks," in *Proceedings of the 28th National Radio Science Conference (NRSC '11)*, April 2011.

[6] A. Dent and Y. Zheng, *Practical Signcryption*, Springer, Berlin, Germany, 2010.

[7] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption), advances," in *Proceedings of the Advances in Cryptology (CRYPTO '97)*, vol. 1294 of *Lecture Notes in Computer Science*, pp. 165–179, Springer, 1997.

[8] R. Steinfeld and Y. Zheng, "A signcryption scheme based on integer factorization," in *Proceedings of the Information Security Workshop (ISW '00)*, vol. 1975 of *Lecture Notes in Computer Science*, pp. 308–322, Springer, 2000.

[9] J. Malone-Lee and W. Mao, "Two birds one stone: signcryption using RSA," in *Proceedings of the Cryptographers' Track at the RSA Conference (CTRSA '03)*, vol. 2612 of *Lecture Notes in Computer Science*, pp. 211–225, Springer, 2003.

[10] Y. Zheng and H. Imai, "How to construct efficient signcryption schemes on elliptic curves," *Information Processing Letters*, vol. 68, no. 5, pp. 227–233, 1998.

[11] M. Toorani and A. A. Beheshti, "A directly public verifiable signcryption scheme based on elliptic curves," in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC '09)*, pp. 713–716, IEEE Computer Society, July 2009.

[12] L. Zhang and T. Mo, "A signcryption scheme for WEP in WLAN based on bilinear pairings," in *Proceedings of the International Conference on Computer Application and System Modeling (ICCASM '10)*, vol. 8, pp. 126–130, IEEE Computer Society, October 2010.

[13] J. Zhang, Y. Yang, and X. Niu, "A novel identity-based multi-signcryption scheme," *International Journal of Distributed Sensor Networks*, vol. 1, no. 5, p. 28, 2009.

[14] F. Li, F. Muhaya, M. Khan, and T. Takagi, "Lattice-based signcryption," *Concurrency and Computation*, vol. 2, pp. 1–10, 2012.

[15] F. Wang, Y. Hu, and C. Wang, "Post-quantum secure hybrid signcryption from lattice assumption," *Applied Mathematics and Information Sciences*, no. 6, pp. 23–28, 2012.

[16] K. Ko, S. Lee, J. Cheon, and J. Han, "New public-key cryptosystem using braid groups," in *Proceedings of the Advances in Cryptology (CRYPTO '00)*, vol. 1880 of *Lecture Notes in Computer Science*, pp. 166–183, Springer, Berlin, Germany, 2000.

[17] I. Anshel, M. Anshel, B. Fisher, and D. Goldfeld, "New key agreement protocols in braid group cryptography," in *The Cryptographers' Track at RSA Conference (CT-RSA '01)*, vol. 2020 of *Lecture Notes in Computer Science*, pp. 13–27, Springer, Berlin, Germany, 2001.

[18] I. Anshel, M. Anshel, and D. Goldfeld, "An algebraic method for public-key cryptography," *Mathematical Research Letters*, vol. 6, no. 3-4, pp. 287–291, 1999.

[19] M. Anshel, "Braid group cryptography and quantum cryptoanalysis," in *Proceedings of the 8th International Wigner Symposium*, pp. 13–27, GSUCCUNY, May 2003.

[20] K. Ko, D. Choi, M. Cho, and J. Lee, "New signature scheme using conjugacy problem," Preprint, 2002, http://eprint.iacr.org/2002/168.

[21] L. Wang, Z. Cao, P. Zeng, and X. Li, "One-more matching conjugate problem and security of braid-based signatures," in *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications security (ASIACCS '07)*, pp. 295–301, ACM Press, March 2007.

[22] L. Wang, Z. Cao, S. Zheng, X. Huang, and Y. Yang, "Transitive signatures from braid groups," in *Proceedings of the Progress in Cryptology (INDOCRYPT '07)*, vol. 4859 of *Lecture Notes in Computer Science*, Springer, December 2007.

[23] L. Wang, L. Wang, Z. Cao, Y. Yang, and X. Niu, "Conjugate adjoining problem in braid groups and new design of braid-based signatures," *Science in China, Series F*, vol. 53, no. 3, pp. 524–536, 2010.

[24] L. Wang, L. Wang, Z. Cao, E. Okamoto, and J. Shao, "New constructions of public-key encryption schemes from conjugacy search problems," in *Proceedings of the International Conference Information Security and Cryptology (Inscrypt '11)*, vol. 6584 of *Lecture Notes in Computer Science*, pp. 1–17, Springer, 2011.

[25] V. Shpilrain and A. Ushakov, "An authentication scheme based on the twisted conjugacy problem," in *Proceedings of the Applied Cryptography and Network Security (ACNS '08)*, vol. 5037 of *Lecture Notes in Computer Science*, pp. 366–372, Springer, Berlin, Germany, 2008.

[26] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in garside groups—I: cyclings, powers, and rigidity," *Groups, Geometry and Dynamics*, vol. 1, no. 3, pp. 221–279, 2007.

[27] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in Garside groups—III: periodic braids," *Journal of Algebra*, vol. 316, no. 2, pp. 746–776, 2007.

[28] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in garside groups II: structure of the ultra summit set," *Groups, Geometry and Dynamics*, vol. 2, no. 1, pp. 16–31, 2008.

[29] K. H. Ko, J. W. Lee, and T. Thomas, "Towards generating secure keys for braid cryptography," *Designs, Codes, and Cryptography*, vol. 45, no. 3, pp. 317–333, 2007.

[30] M. Prasolov, "Small braids having a big ultra summit set," http://arxiv.org/abs/0906.0076.

[31] P. Dehornoy, "Braid-based cryptography," *Contemporary Mathematics—American Mathematical Society*, vol. 360, pp. 5–33, 2004.

[32] P. Dehornoy, "Using shited conjugacy in braid-based cryptography," *Algebraic Methods in Cryptography, Contemporary Mathematics—American Mathematical Society*, vol. 418, pp. 65–74, 2006.

[33] U. Maurer, "Abstract models of computation in cryptography," in *Proceedings of the Cryptography and Coding*, N. P. Smart, Ed., vol. 3796 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, Heidelberg, Germany, 2005.

[34] E. Fujisaki and T. Okamoto, "How to enhance the security of public-key encryption at minimum cost," in *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC '99)*, H. Imai and Y. Zheng, Eds., vol. 1560 of *Lecture Notes in Computer Science*, pp. 53–68, Springer, Heidelberg, Germany, 1999.

[35] L. Gu, L. Wang, K. Ota, M. Dong, Z. Cao, and Y. Yang, "New public key cryptosystems based on non-abelian factorization problems," *Security and Communication Networks*. In press.

[36] D. Kahrobaei and C. Koupparis, "Non-commutative digital signatures," *Groups Complexity and Cryptology*, vol. 4, pp. 377–384, 2012.

[37] A. Dent, "Hybrid signcryption schemes with insider security," in *Proceedings of the 10th Australasian Conference on Information Security and Privacy (ACISP '05)*, vol. 3574 of *Lecture Notes in Computer Science*, pp. 253–266, Springer, 2005.

[38] A. Dent, "Hybrid signcryption schemes with outsider security," in *Proceedings of the 8th International Conference on Information Security (ISC '05)*, vol. 3650 of *Lecture Notes in Computer Science*, pp. 203–217, Springer, 2005.

[39] J. Cha, K. Ko, S. Lee et al., "An efficient implementation of braid groups," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT '01)*, vol. 2248 of *Lecture Notes in Computer Science*, pp. 144–156, Springer, Berlin, Germany, 2001.

[40] S. Maffre, "A weak key test for braid based cryptography," *Designs, Codes, and Cryptography*, vol. 39, no. 3, pp. 347–373, 2006.

[41] D. Coppersmith, "Modifications to the number field sieve," *Journal of Cryptology*, vol. 6, no. 3, pp. 169–180, 1993.

[42] J. Hughes, "The left sss attack on ko-lee-cheon-han-kang-park key agreement scheme in b_{45}," in *Rump Session Crypto*, 2000.

[43] J. Hughes, "A linear algebraic attack on the aafg1 braid group cryptosystem," in *Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP '02)*, vol. 2384 of *Lecture Notes in Computer Science*, pp. 176–189, Springer, Berlin, Germany, 2002.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 138384, 10 pages
http://dx.doi.org/10.1155/2013/138384

Research Article

A Cross-Layer Security Scheme of Web-Services-Based Communications for IEEE 1451 Sensor and Actuator Networks

Jun Wu,[1,2] Ming Zhan,[3] Bin Duan,[4] and Jiang Liu[5]

[1] *Research Institute for Secure Systems (RISEC), National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8568, Japan*
[2] *Global Information and Telecommunication Institute (GITI), Waseda University, Tokyo 169-0051, Japan*
[3] *School of Electronic and Information Engineering, Southwest University, Chongqing 400714, China*
[4] *College of Information Engineering, Xiangtan University, Xiangtan 411105, China*
[5] *School of Electrical and Electronics Engineering, North China Electric Power University, Beijing 102206, China*

Correspondence should be addressed to Ming Zhan; zmdjs@swu.edu.cn

Received 17 December 2012; Accepted 1 February 2013

Academic Editor: Mianxiong Dong

IEEE 1451 standard has been proposed to provide a common communication interface and transducer electric data sheet format for wired and wireless distributed applications in smart transducers (sensors and actuators). Currently, a unified Web service for IEEE 1451 smart transducers is a must. However, ensuring the security of web-services-based communications for IEEE 1451 smart transducers is an unsolved problem. In this paper, we proposed a cross-layer security mechanism that deals with the requirements of authentication, integrity, confidentiality, and availability across the communication process in IEEE 1451 smart transducers. The scheme contains three cross-layer components logically, including XML Encryption and Signature, SOAP Security Extension, and Web Services Description Language (WSDL) Security Checking. The former two components satisfy the requirements of confidentiality, availability, integrity, authentication, nonrepudiation, and freshness. The third component satisfies the requirement of availability, which can protect the system against denial-of-service (DoS) attack. The three cross-layer security components are integrated seamlessly in our scheme. To evaluate the overhead, we perform tests to evaluate the effect of message size on the performance of the access inquiry web service. The result supports the usefulness and feasibility of our scheme.

1. Introduction

In recent years, sensor and actuator have attracted a lot of attention recently due to their broad applications, ranging from industrial automation to environmental condition monitoring and control-to-intelligent transportation system to homeland defense [1–5]. A smart transducer is a compact unit containing a sensor or actuator element, a microcontroller, a communication controller, and the associated software for signal conditioning, calibration, diagnostics, and communication [6–8]. A smart transducer can enable novel application in and beyond measurement, monitoring, control, and actuating [9].

The behaviors to smart transducers generally call for distributed and remote architecture [10–12]. And these systems usually require a variety of networked interconnections and telecommunication technologies for measurement and control, and the devices are usually made by different manufactures. Therefore, common and reliable communication interface and data format are important for smart transducers. As a consequence, the Instrumentation and Measurement Society's Sensor Technology Technical Committee TC-9 at the Institute of Electrical and Electronics Engineers (IEEE) has been working to establish a group of smart sensor interface standards called IEEE 1451 [13–18]. IEEE 1451 standard is proposed to provide a common communication interface and transducer electric data sheet format for wired and wireless distributed applications. It will eliminate the issue of proprietary communication systems utilizing a wide variety of protocols, labels, semantics, and so forth, thus enabling a transducer application to exchange information with different smart transducers independently of a vendor.

From a utility perspective, unified definitions of common data minimize conversion and recalculation of data values for evaluation and comparison in many application systems.

Recently, the working group of Kang Lee, who is the Chairman of the IEEE Instrumentation and Measurement Society's Technical Committee on Sensor Technology and responsible for the establishment of the suite of IEEE 1451, proposed a unified Web service for IEEE 1451 smart transducers [19]. This work developed the IEEE 1451 standard to a new emerging unified Web service framework. An IEEE 1451 Network Capable Application Processor (NCAP) can be used as a Smart Transducer Web Services (STWS) provider, which provides asset of Web services for the STWS. STWS consumers, such as sensor alert system, OGC-SWE, or other applications, can find the STWS deployed and then invoke the STWS through Simple Object-access Protocol (SOAP)/Extensible Markup Language (XML) message. As a consequence, the use of Web service technologies provides the benefits of low implementation cost and ease of interoperability because Web services can implement service-oriented architectures (SOAs), which enable loosely coupled integration and interoperation of distributed heterogeneous system by using services as component elements in transducer networks. However, on the other hand, Web-services-based communication introduces the cyber security problem.

The importance of cyber security in sensor and actuator networks is widely recognized. Recently, schemes related to the cyber security for sensor and actuator networks have been widely investigated [20–24]. In particular, cyber security of Web-services-based communication for smart transducers must be implemented [25]. Security issues of communication for smart transducer are described in IEEE 1451.0. However, how the security issues are handled is up to the individual supplier and the responsibilities of communication protocol [13]. As a matter of fact, Web-services-based communication for smart transducers is a new emerging technology, in which few studies have been conducted for security. A common method of implement security is based on a secure transport layer or network layer, which typically includes secure socket layer (SSL), transport layer security (TLS), and network layer security (NLS). For example, TLS and NLS are recommended to secure TCP/IP-based communication for wireless sensor and actuator networks in IEEE 1415.5. However, these security schemes provide security only in a secure channel, and not in files or databases. Furthermore, these techniques do not correspond with the web services architecture in which the intermediaries can manipulate the messages on their way. Once using a secure transport layer, intermediaries are not able to control the message [26, 27].

The Web Services Security (WS-Security) [28, 29] standard was produced by Advancing Open Standards for the Information Society (OASIS) in 2004. The Web Services Security (WS-Security) is an essential component of the Web services protocol stack to provide end-to-end integrity, confidentiality, and authentication capabilities to web services. End-to-end message security assures the participation of nonsecure transport intermediaries in message exchanges, which is a key advantage for web systems and service-oriented architectures. Some security schemes corresponding with WS-Security are proposed for e-mail system, enterprise services system, trust management, and so forth, but cannot be applied directly to smart transducers [28–31].

As a matter of fact, IEEE 1451 standards define a common communication interfaces for networked smart transducers, which include sensors and actuators. The research of sensor and actuator networks is an existing area. In this paper, IEEE 1451 sensor and actuator networks means the networked smart transducers which are based on the common interfaces of IEEE 1451. Because of the communication protocols and data format of IEEE 1451 sensor and actuator networks, the secure communication proposals should have their special features based on IEEE 1451 standards. On the other hand, this paper focuses on the Web services communication security. So the security topics of confidentiality, availability, integrity, authentication, nonrepudiation, and freshness are necessary for the IEEE 1451 sensor and actuator networks, which also must be based on the data format of IEEE 1451.

In this paper, we proposed a cross-layer security scheme for web-services-based communication for IEEE 1451 smart transducers. The rest of this paper is organized as follows. Section 2 analyzes the system architecture and security requirements of IEEE 1451 reference model. Section 3 presents the architecture and security measures of web services. Section 4 presents the proposed security scheme. Section 5 analyzes the security of the proposed scheme. Section 5 evaluates the implementation of the proposed scheme. Finally, Section 6 concludes this paper.

2. System Architecture and Security Requirements

2.1. IEEE 1451 Reference Model. IEEE 1451 standards define a common communication model to connect smart transducers to normalization of integrated, intelligent, and open distributed measurement and control systems (DMCSs). Figure 1 shows the IEEE 1451 reference model. The IEEE 1451 family of standards divides the parts of a smart transducer system into two general categories of devices. One is the Network Capable Application Processor (NCAP) that functions as a gateway between the users' network and the transducer interface modules (TIMs) [13–18]. In the IEEE 1451 reference model, smart transducers connect with DMCS users through the user communication network. The user communication network is outside of the scope of the IEEE 1451 family of standards. It may be anything that the user desires. The only requirement that is placed on the NCAP is that the NCAP has the appropriate network interface hardware and software [13].

The communications between NCAP and TIM are based on IEEE 1451.X communication modules in both sides, which provide the low levels of the communications protocol [13]. DMCS users interact with smart transducers through public application programming interfaces (APIs) [13]. The applications run in NCAP or remote DMCS system interact with transducers through public application programming interfaces (APIs).

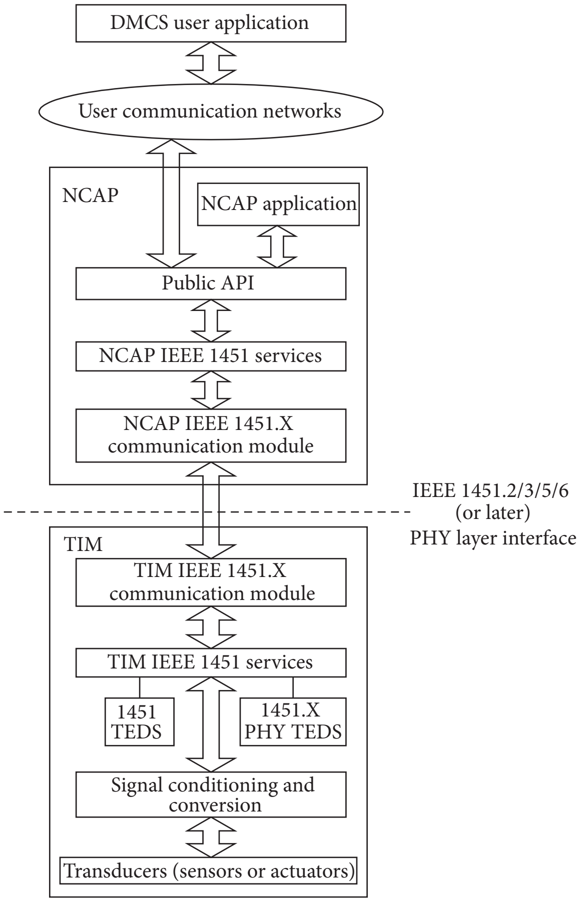

DMCS user application
User communication networks
NCAP
NCAP application
Public API
NCAP IEEE 1451 services
NCAP IEEE 1451.X communication module
IEEE 1451.2/3/5/6 (or later) PHY layer interface
TIM
TIM IEEE 1451.X communication module
TIM IEEE 1451 services
1451 TEDS
1451.X PHY TEDS
Signal conditioning and conversion
Transducers (sensors or actuators)

Figure 1: IEEE 1451 reference model.

2.2. Unified Web Service for IEEE 1451 Smart Transducers. Web services are typically APIs or web APIs that are accessed via Hypertext Transfer Protocol and executed on a remote system hosting the requested services. Qualities like simplicity, code reuse, and interoperability are also making Web services a *de facto* standard in the context of DMCSs [32]. The IEEE 1451 working group proposed a unified Web service for IEEE 1451 smart transducers recently [19]. In fact, in the reference model in Figure 1, the NCAP application module is logically an optional complement to provide the functions and service to pass information across the interface between the DMAC users and NCAP [33]. For mapping to Web services, the method introduced in [19] designed smart transducer Web services (STWSs) in the NCAP application services component. Hence, an IEEE 1451 NCAP provides a set of Web services for the STWS, which acts as an STWS provider. As shown in Figure 1, a common basis for the members of the IEEE 1451 family of standards is provided to be interoperable [13]. Hence, STWSs based on the IEEE 1451.0 standard have been defined using Web Services Description Language (WSDL). The DMCS user applications and STWS provider communicate with each other using SOAP/XML messages. The communications between NCAP and TIM are based on IEEE 1451.X [19].

2.3. Security Requirements. In this paper, we consider the security of the communication between the DMCS users and IEEE 1451 smart transducers purposely.

The security issues of the communication in the smart transducers are the responsibilities of IEEE 1451.X but not IEEE 1451.0 [13], and security in IEEE 1451.X is based on the specified communication protocol, such as bluetooth in IEEE 1451.5. However, the security of the communication between STWS consumers and STWS providers is an unsolved problem, which should be designed based on IEEE 1451.0 combined with Web services.

Recently, the security requirements of data exchange in sensor and actuator networks have been widely discussed, which include [34–37]:

(i) confidentiality: confidentiality or secrecy has to do with making information inaccessible to unauthorized users. A confidential message is resistant to revealing its meaning to an eavesdropper.

(ii) Availability: availability ensures the survivability of network services to authorized parties when needed despite denial-of-service (DoS) attacks. A denial-of-service attack could be launched at any OSI (open system interconnect) layer of a sensor network.

(iii) Integrity: integrity measures ensure that the received data is not altered in transit by an adversary.

(iv) Authentication: authentication enables a node to ensure the identity of the peer node with which it is communicating.

(v) Nonrepudiation: nonrepudiation denotes that a node cannot deny sending a message it has previously sent.

(vi) Access control: access control implement the process of identifying nodes as well as authorizing and granting nodes the access right to information or resources.

(vii) Freshness: this could mean data freshness and key freshness. Since all sensor networks provide some forms of time-varying measurements, we must ensure that each message is fresh. Data freshness implies that each data is recent, and it ensures that no adversary replayed old messages.

The above requirements are in conformance with the security requirements of data exchange described in ISO/IEC 29180 working draft [38], which is standard under development for security framework for ubiquitous sensor network.

In the above security requirements, access control can be performed based on access control scheme. The security access control scheme introduced in [23] is useful for IEEE 1451 smart sensors.

Other security requirements, which are confidentiality, availability, integrity, authentication, nonrepudiation, and freshness, should be implemented based on IEEE 1451.0 integrated with Web services.

IEEE 1451 standards defines six transducer services [13, 19, 33], which are *TimDiscovery*, *TransducerAccess*, *TransducerManager*, *TedsManager*, *CommManager*, and *AppCallback*.

Table 1 shows the security requirements of communication of the responding services.

Table 1: Security requirements of communication process.

Service	Security requirements for the communications
TimDiscovery	Availability, integrity, nonrepudiation, and freshness
TransducerAccess	Confidentiality, availability, integrity, authentication, nonrepudiation, and freshness
TransducerManager, TedsManager, and CommManager	Confidentiality, availability, integrity, authentication, nonrepudiation, and freshness
AppCallback	Integrity, nonrepudiation, and freshness

3. Web Services Architecture and Web Services Security

3.1. Web Services Architecture. Today, the ability to seamlessly exchange information between internal business units, customers, and partners is vital for success; yet most organizations employ a variety of disparate applications that store and exchange data in dissimilar ways and therefore cannot "talk" to one another productively. Web services have evolved as a practical, cost-effective solution for uniting information distributed between critical applications over operating system, platform, and language barriers that were previously impassable.

Web services [39] are in simple terms object methods exposed via HTTP using pure SOAP messages. The major components or layers of a Web Service Protocol Stack include

(1) Extansible Markup Language (XML) layer: providing a means for communicating over the Web using an XML document that both requests and responds to information between two disparate systems.

(2) Simple Object Access Protocol (SOAP) layer: a XML Messaging specification, which allows the sender and the receiver of XML documents to support a common data transfer protocol for effective networked communication.

(3) Web Services Description Language (WSDL) layer: playing an important role in the overall Web services architecture since it describes the complete contract for application communication.

(4) Universal Description, Discovery and Integration (UDDI) layer: a platform-independent, Extensible Markup Language- (XML-) based registry, which represents a way to publish and find web services over the Web.

Figure 2 shows the protocol stack architecture of Web Services.

3.2. Web Services Security. Web Services Security is based on open W3C-approved XML standards [40, 41], which provide the security foundation for applications of Web services. The standards are platform neutral, thus promoting interoperability. Also, OASIS published the standards for defining the security expanding method for SOAP message exchange [42].

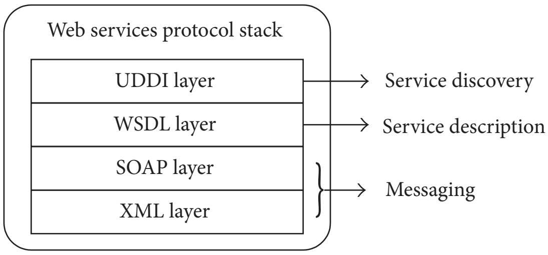


Figure 2: Web services protocol stack.

4. The Proposed Security Scheme

4.1. Basic Idea and Model. The basic idea and model of the proposed security scheme are shown in Figure 3. The goal of the security scheme is to satisfy the security requirements of the data exchange. The proposed approach can be viewed as "cross-layer design" at the messaging layer and description layer in Web Services protocol stack. The scheme contains three components logically, including XML Encryption and Signature, SOAP Security Extension, and WSDL Security Checking. The former two components satisfy the requirements of confidentiality, availability, integrity, authentication, nonrepudiation, and freshness. The third component satisfies the requirement of availability, which can protect the system against DoS attack. For mapping to Web Services, the security scheme is designed based on the layer architecture of Web services protocol stack. Also, the scheme is designed in conformance with the Web services security standard. Most important, the three components of the security scheme are based on IEEE 1415 transducer services, services API, and XML schema of API, respectively, which are defined in IEEE 1451 standards. As described in IEEE 1451.0, all text strings in the Transducer Electronic Data Sheet (TEDS) shall conform to W3C Recommendation Extensible Markup Language (XML) 1.0 (Second Edition).

4.2. Secure XML Messaging Layer. A security token represents a collection of claims, which is used to prove one's identity and provide the foundation for ensuring the confidentiality, integrity, nonrepudiation, and freshness of the data. Web Services Security standard defined several security tokens, including X. 509 certificate token, username/password token, Kerberos token, and SAML token. The security token most commonly used in DMCSs and sensor networks is username/password token [43–48].

Table 2 lists the notations used throughout the description of the security scheme for ease of reference.

4.2.1. Secure Messages of TimDiscovery. Figure 8 shows the protocol of secure message of *TimDiscovery*. For generating the signature, the client node first generates a fresh nonce R_U. Then, she computes the digital digest of her own password

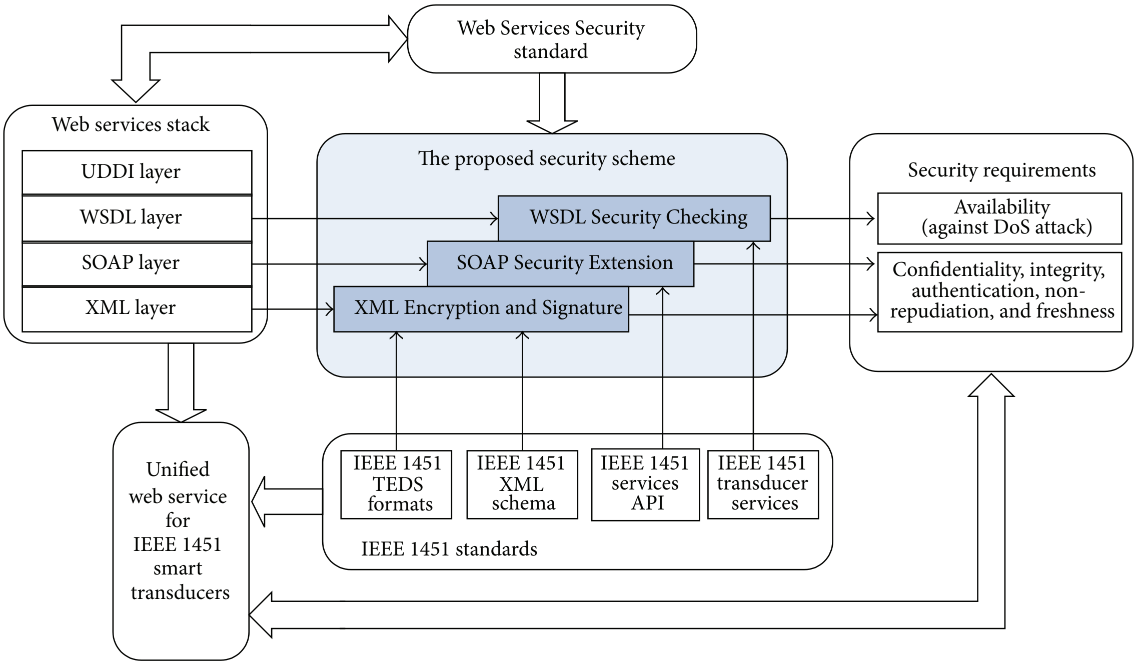

Web Services Security standard
Web services stack
UDDI layer
WSDL layer
SOAP layer
XML layer
The proposed security scheme
WSDL Security Checking
SOAP Security Extension
XML Encryption and Signature
Security requirements
Availability (against DoS attack)
Confidentiality, integrity, authentication, non-repudiation, and freshness
Unified web service for IEEE 1451 smart transducers
IEEE 1451 TEDS formats
IEEE 1451 XML schema
IEEE 1451 services API
IEEE 1451 transducer services
IEEE 1451 standards

Figure 3: Basic idea and model.

together with the fresh nonce based on hash based message authentication code (HMAC). The password is stored in both the memory of client node and the server node. Clearly, the signature provides a nonrepudiation property. This is true because only the client node herself can generate it, and the fresh nonce guarantees its freshness. Next, nonce and created time are the additional elements to resist against the replay attack. Then, the client node generates a signature *RequestMAC*, which is for *RequestParameters*. *RequestParameters* is the original message of the access request. Next, the client node sends out *Username*, *ReqParameters*, T, R_U, and *RequestMAC*. After receiving the message from the client node, the server node retrieves the parameters from the message. Then, S computes the *RequestMAC'* based on the parameters from the message. After that, S verifies the signature through comparing *RequestMAC* and *RequestMAC'*. Then, a symmetric key is derived based on the password and a 16-bit random value G. Next, S computes the signature of response and the symmetric key. These values then are sent back to U. After U gets the message, U can derive the symmetric key.

4.2.2. Secure Messages of TransducerAccess, TransducerManager, TedsManager, and CommManager. Figure 9 shows the protocol of secure message of *TransducerAccess*, *TransducerManager*, *TedsManager*, and *CommManager*. The client node U firstly generates a signature, which includes the generation of a fresh nonce R_U and the computation of the digital digest of her own password together with the fresh nonce based on HMAC. As a matter of fact, the generation process of the signature in this section is similar to the process of *TimDiscovery*.

Table 2: Notation used by the secure authentication protocol.

Notation	Meaning
U	A services consumer
S	A services provider
Key	Shared secret key between U and S for symmetric encryption and decryption
R_A	A nonce generated by entity A, usually it is a randomized value to defend replay attack
T	The created time of message
Salt	A random number to for derive the symmetric key
(M_1, M_2)	Concatenation of two messages
HMAC (M)	Calculate MAC for message M based hash function
H (M)	Apply one-way function to message M
$\{M\}_{Key}$	Encrypt message M by symmetric key algorithm with the secret key between user and service provider
ReqParameters	Parameters involved in the request
ResParameters	Parameters involved in the response

The password is stored in both the memory of client node and the sever node. Next, she generates a security token *ET* based on username/password method. Nonce and created time are the additional elements to resist against the replay attack. Then, she encrypts the *ReqParameters* based on the symmetric key. *ReqParameters* is the original message of the access request. Next, the client node sends out $\{ReqParameters\}_{Key}$, *ET*, and *ResponseMAC*. After receiving the message and the security token from client node, server

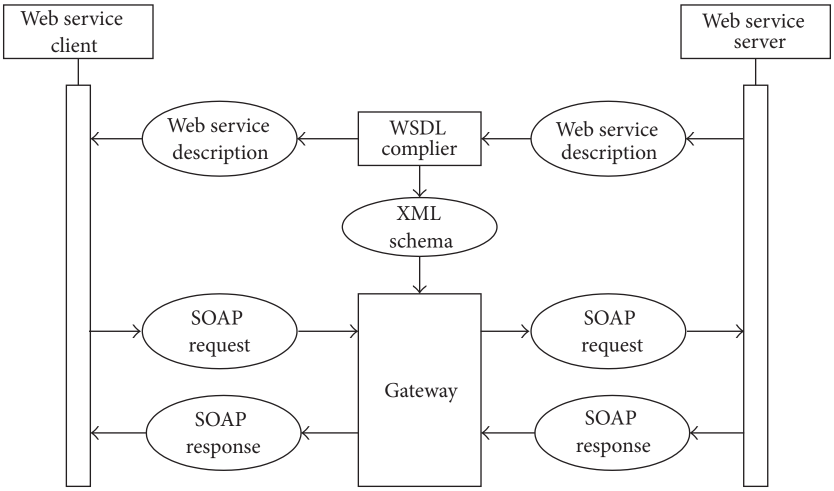


Figure 4: Integration of the CheckWay Web service firewall.

node retrieves the password by corresponding C from the local database, then calculates the PasswordDigest', compares it with PasswordDigest, and authenticates the identity of client node as being equal or not. After verification, the sever node *S* sends an access response message, including a signature of response parameters, *ResParameters*, to ensure the integrity and nonrepudiation. After getting the response message, U will verify the signature of *ResParameters* and then derive *ResParameters* from the message.

4.2.3. Secure Messages of AppCallback. The security requirements of *AppCallback* are as same as those of *TimDiscovery* except that *AppCallback* lacks availability. In the proposed scheme, ensuring availability is the responsibility of the security design of the WSDL layer. In addition, as defined in IEEE 1451.0 standard, *AppCallback* is implemented when applications that need advanced features exist [13]. *Appcallback* is implemented after *TimDiscovery*, which means that the key for symmetric encryption and decryption has already been generated when *AppCallback* is implemented. Hence, at XML messaging layer, the security mechanisms for securing message of *AppCallback* is as same as those of *TimDiscovery* but *key* generation is not needed.

4.3. Secure WSDL Layer. The security design of message layer cannot deal with the requirements of availability because the XML encryption and decryption can only ensure the confidentiality, availability, integrity, authentication, nonrepudiation, and freshness. Current Web services architecture does not consider validation of Web services messages against WSDLs during message processing. This could pose a potential security risk to enterprise servers hosting Web services. We secure the availability at the WSDL layer.

In fact, the most important aspect of a Web service is the service description using the Web Services Description Language (WSDL) that describes the messages, types, and operations of Web service and the contract to which the Web service guarantees it will conform [49]. WSDL plays an important role in the overall Web services architecture since it describes the complete contract for application communication. Smart transducer Web services (STWSs) in [19] are defined using Web services WSDL. WSDL is extensible to allow the description of endpoints and their messages regardless of what message formats or networks protocols are used to communicate. We secure the WSDL layer security based on the method in [50].

The considerations above regarding SOAP message validation lead to the Web service firewall, called CheckWay. Figure 4 shows the integration of a Web service firewall between Web service client and server. The security WSDL compiler gets the Web service server's Web service description, generates the corresponding XML message schema, "hardens" the description, and advertises the modified description to a Web service client. The CheckWay Gateway validates all SOAP messages against the schema, forwards the message if it is valid, and rejects the message if it is not. The next step is now to consider how to obtain an XML schema for the message validation and which problems regarding the firewalls performance emerge from the validation process. In order to answer the first question, a closer look at Web service client/server interaction and the Web service interface description is required. The compiling process is shown in Figure 5.

The SOAP message's structure belonging to a Web service description is defined by information spread all over the description document. The description must be traversed and the information necessary for a specific service or operation must be merged into a message definition.

5. Security Analysis

The basic secure authentication protocol for *TimDiscovery*, *TransducerAccess*, *TransducerManager*, *TedsManager*, *CommManager*, and *Callback* can provide a nonrepudiation property because only the client node herself can generate it, and the fresh nonce guarantees its freshness. Integrity property

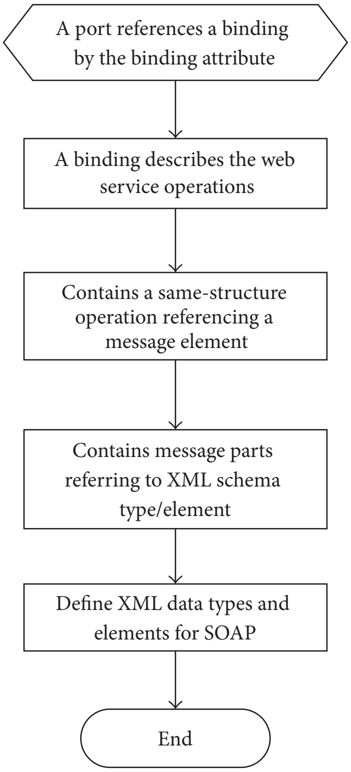

A port references a binding by the binding attribute
A binding describes the web service operations
Contains a same-structure operation referencing a message element
Contains message parts referring to XML schema type/element
Define XML data types and elements for SOAP
End

Figure 5: Compiling a Web service description.

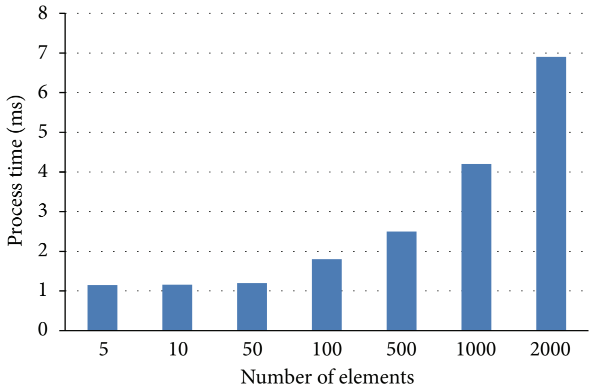


Figure 6: SOAP message length impact on process time.

can also be proved based on HMAC. Moreover, after the nonce and created time are added into the data packet, the receiver can check whether the nonce has been received before or whether the message is created in a very recent time. Thereby, nonce and created time combined into data packets can resist replay attack. Also, we consider the DoS attack model that consists in injecting bogus messages into the system. And before verifying PasswordDigest, only a hash computation needs to be implemented. At the same time, before verifying PasswordDigest, few values need be stored. Therefore, our protocol can resist DoS attack to some extent.

Beside the basic above security, the authentication protocols of *TransducerAccess*, *TransducerManager*, *TedsManager*, *CommManager* use both symmetric encryption and MAC for the message. Therefore, these protocols can provide confidentiality and authentication for the communications.

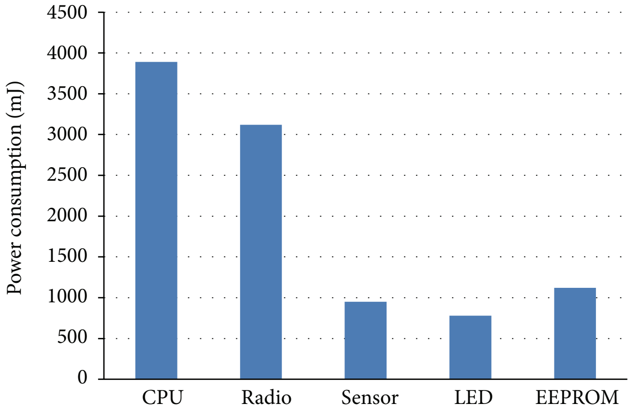


Figure 7: Power consumption on sensor.

6. Implementation

6.1. Time Overhead of Process of CheckWay Gateway. In this section, we present the important aspect of the performance results. We evaluate the effect of message size on the performance of the access inquiry Web services of CheckWay gateway. A laptop is used to simulate the CheckWay gateway, which includes the Intel Core i5 M520 and 2 GB memory. In the implementation, we varied the message size by increasing the number of XML elements contained in the response message. As shown in Figure 6, the time consumption of CPU required to process an account inquiry request depends on the number of elements returned in the response message. The longest message is 400 times larger than the smallest message, but the increase in CPU consumption is less than 5-fold. Please note that in the implementation a SOAP message of 50 elements contains 1 KB of data, but the message itself has a length of 2.3 KB because of the XML tags.

6.2. Power Consumption of Sensor. It is very important to verify the feasibility of the implementation of the proposed scheme on resource-constrained sensors. In this subsection, we estimate the energy consumption of sensor using PowerTOSSIM [51], which is an energy modeling extension of TOSSIM for the simulation of MICAz mote. Here, we take *TimDiscovery* message authentication as the example for evaluation. The energy consumption is measured for five components: CPU, RADIO, LED, SENSOR, and EEPROM. We fix the time of execution equal to 1200 simulated seconds, which is because the motes in PowerTOSSIM take boot time of 10 seconds. In our scheme, storing security data performed by EEPROM component and computations performed by CPU component slightly increase the energy consumption, where radio transmission is not always necessary and accordingly the RADIO component energy consumption is greatly reduced. As shown in Figure 7, the energy consumption of our scheme is acceptable for resource-constrained WSNs.

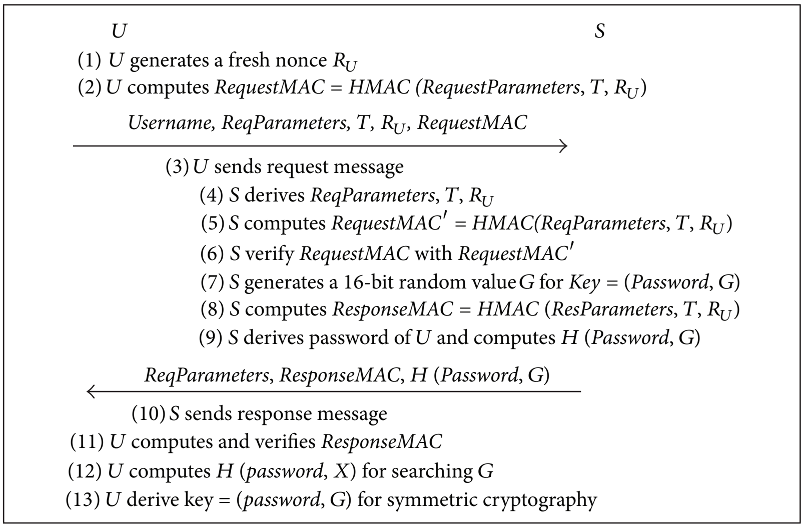

U
S
(1) U generates a fresh nonce R_U
(2) U computes $RequestMAC = HMAC\ (RequestParameters, T, R_U)$
$Username, ReqParameters, T, R_U, RequestMAC$
(3) U sends request message
(4) S derives $ReqParameters, T, R_U$
(5) S computes $RequestMAC' = HMAC(ReqParameters, T, R_U)$
(6) S verify $RequestMAC$ with $RequestMAC'$
(7) S generates a 16-bit random value G for $Key = (Password, G)$
(8) S computes $ResponseMAC = HMAC\ (ResParameters, T, R_U)$
(9) S derives password of U and computes $H\ (Password, G)$
$ReqParameters, ResponseMAC, H\ (Password, G)$
(10) S sends response message
(11) U computes and verifies $ResponseMAC$
(12) U computes $H\ (password, X)$ for searching G
(13) U derive key = $(password, G)$ for symmetric cryptography

Figure 8

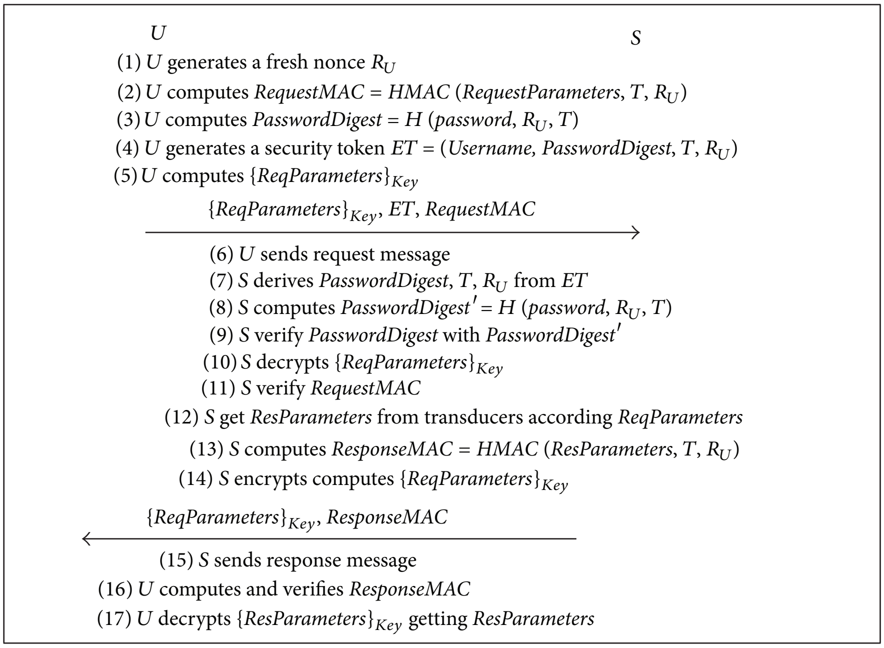

U
S
(1) U generates a fresh nonce R_U
(2) U computes $RequestMAC = HMAC\ (RequestParameters, T, R_U)$
(3) U computes $PasswordDigest = H\ (password, R_U, T)$
(4) U generates a security token $ET = (Username, PasswordDigest, T, R_U)$
(5) U computes $\{ReqParameters\}_{Key}$
$\{ReqParameters\}_{Key}, ET, RequestMAC$
(6) U sends request message
(7) S derives $PasswordDigest, T, R_U$ from ET
(8) S computes $PasswordDigest' = H\ (password, R_U, T)$
(9) S verify $PasswordDigest$ with $PasswordDigest'$
(10) S decrypts $\{ReqParameters\}_{Key}$
(11) S verify $RequestMAC$
(12) S get $ResParameters$ from transducers according $ReqParameters$
(13) S computes $ResponseMAC = HMAC\ (ResParameters, T, R_U)$
(14) S encrypts computes $\{ReqParameters\}_{Key}$
$\{ReqParameters\}_{Key}, ResponseMAC$
(15) S sends response message
(16) U computes and verifies $ResponseMAC$
(17) U decrypts $\{ResParameters\}_{Key}$ getting $ResParameters$

Figure 9

7. Conclusion

To secure the web-services-based communications for networked IEEE 1451 smart transducers, we proposed a cross-layer security mechanism, which is based on the layer architecture of Web services protocol stack. The security requirements are derived from IEEE 1451 and Web service communications, and the design is consistent with existing applications of IEEE 1451 web services communication utilities and an information security standard. Moreover, the scheme is designed in conformance with the Web Services Security standard. Most important, the three components of the security scheme are based on IEEE 1451 transducer services, services API, and XML schema of API, respectively, which are defined in IEEE 1451 standards. The effect of message size on the performance of the access inquiry web service is tested, which verifies the feasibility of our scheme. The proposed scheme provides an efficient reference security model of web-services-based communications for networked IEEE 1451 smart transducers.

Acknowledgment

The authors would like to thank the guest editor and anonymous reviewers for their helpful and constructive comments.

References

[1] E. Y. Song and K. Lee, "Understanding IEEE 1451—networked smart transducer interface standard—what is a smart transducer?" *IEEE Instrumentation and Measurement Magazine*, vol. 11, no. 2, pp. 11–17, 2008.

[2] K. Ota, M. Dong, and X. Li, "TinyBee: mobile-agent-based data gathering system in wireless sensor networks," in *Proceedings of the IEEE International Conference on Networking, Architecture, and Storage (NAS'09)*, pp. 24–31, Hunan, China, July 2009.

[3] J. Liang, X. Zeng, W. Wang, and H. Chen, "L-shaped array-based elevation and azimuth direction finding in the presence of mutual coupling," *Signal Processing*, vol. 91, no. 5, pp. 1319–1328, 2011.

[4] S. R. Rossi, A. A. de Carvalho, A. C. R. da Silva et al., "Open and standardized resources for smart transducer networking," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 10, pp. 3754–3761, 2009.

[5] B. Liu, H. Chen, Z. Zhong, and H. V. Poor, "Asymmetrical round trip based synchronization-free localization in large-scale underwater sensor networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3532–3542, 2010.

[6] K. B. Lee and E. Y. Song, "Object-oriented application framework for IEEE 1451.1 standard," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 4, pp. 1527–1533, 2005.

[7] M. Dong, K. Ota, X. Li, X. Shen, S. Guo, and M. Guo, "HARVEST: a task-objective efficient data collection scheme in wireless sensor and actor networks," in *Proceedings of the 3rd International Conference on Communications and Mobile Computing (CMC'11)*, pp. 485–488, Qingdao, China, April 2011.

[8] L. Chen, W. Chen, B. Wang, X. Zhang, H. Chen, and D. Yang, "System-level simulation methodology and platform for mobile cellular systems," *IEEE Communications Magazine*, vol. 49, no. 7, pp. 148–155, 2011.

[9] H. Chen, G. Wang, Z. Wang, and H. So, "Non-line-of-sight node localization based on semi-definite programming in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 108–116, 2012.

[10] M. Staroswiecki, "Intelligent sensors: a functional view," *IEEE Transactions on Industrial Informatics*, vol. 1, no. 4, pp. 238–249, 2005.

[11] K. Ota, M. Dong, J. Wang, S. Guo, Z. Cheng, and M. Guo, "Dynamic itinerary planning for mobile agents with a content-specific approach in wireless sensor networks," in *Proceedings of the 72nd IEEE Vehicular Technology Conference Fall (VTC2010-Fall)*, pp. 1–5, Ottawa, Canada, September 2010.

[12] G. Wang and H. Chen, "An importance sampling method for TDOA-based source localization," *IEEE Transactions on Wireless Communications*, vol. 10, no. 5, pp. 1560–1568, 2011.

[13] "IEEE standard for a smart transducer interface for sensors and actuators—common functions, communication protocols, and transducer electronic data sheet (TEDS) formats," IEEE Standards Board, IEEE Std 1451. 0-2007, 2007.

[14] "IEEE standard for a smart transducer interface for sensors and actuators—network capable application processor (NCAP) information model," IEEE Standards Board, IEEE Std 1451. 1-1999, 1999.

[15] "IEEE standard for a smart transducer interface for sensors and actuators—transducer to microprocessor communication protocols and transducer electronic data sheet (TEDS) formats," IEEE Standards Board, IEEE Std 1451. 2-1997, 1997.

[16] "IEEE standard for a smart transducer interface for sensors and actuators—digital communication and transducer electronic data sheet (TEDS) formats for distributed multidrop system," IEEE Standards Board, IEEE Std 1451. 3-2003, 2003.

[17] "IEEE standard for a smart transducer interface for sensors and actuators—wireless communication protocols and transducer electronic data sheet (TEDS) formats," IEEE Standards Board, IEEE Std 1451. 5-2007, 2007.

[18] "IEEE standard for a smart transducer interface for sensors and actuators - Transducers to radio frequency identification (RFID) systems communication protocols and transducer electronic data sheet formats," IEEE Standards Board, IEEE Std 1451. 7-2010, 2010.

[19] E. Y. Song and K. B. Lee, "STWS: a unified web service for IEEE 1451 smart transducers," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 8, pp. 1749–1756, 2008.

[20] B. Panja, S. K. Madria, and B. Bhargava, "A role-based access in a hierarchical sensor network architecture to provide multilevel security," *Computer Communications*, vol. 31, no. 4, pp. 793–806, 2008.

[21] L. Maccari, L. Mainardi, M. A. Marchitti, N. R. Prasad, and R. Fantacci, "Lightweight, distributed access control for wireless sensor networks supporting mobility," in *Proceedings of the IEEE International Conference on Communications (ICC'08)*, pp. 1441–1445, Beijing, China, May 2008.

[22] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 3–13, 2007.

[23] J. Wu and S. Shimamoto, "Usage control based security access scheme for wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC'10)*, May 2010.

[24] K. Ota, M. Dong, Z. Cheng, J. Wang, X. Li, and X. Shen, "ORACLE: mobility control in wireless sensor and actor networks," *Computer Communications*, vol. 35, no. 9, pp. 1029–1037, 2012.

[25] K. B. Lee and M. E. Reichardt, "Open standards for homeland security sensor networks—sensor interconnection and integration trough Web access," *IEEE Instrumentation and Measurement Magazine*, vol. 8, no. 5, pp. 14–21, 2005.

[26] J. Viega and J. Epstein, "Why applying standards to web services is not enough," *IEEE Security and Privacy*, vol. 4, no. 4, pp. 25–31, 2006.

[27] E. Kleiner and A. W. Roscoe, "On the relationship between web services security and traditional protocols," *Electronic Notes in Theoretical Computer Science*, vol. 155, no. 1, pp. 583–603, 2006.

[28] Oasis Consortium, WS-Security specification, 2004, https://www.oasis-open.org/.

[29] Z. Wu and A. C. Weaver, "Using web services to exchange security tokens for federated trust management," in *Proceedings of the IEEE International Conference on Web Services (ICWS'07)*, pp. 1176–1178, Salt Lake City, Utah, USA, July 2007.

[30] M. Anlauff, D. Pavlovic, and A. Suenbuel, "Deriving secure network protocols for enterprise services architectures," in *Proceedings of the IEEE International Conference on Communications (ICC'06)*, pp. 2283–2287, Istanbul, Turkey, July 2006.

[31] L. Liao and J. Schwenk, "Secure emails in XML format using web services," in *Proceedings of the 5th IEEE European Conference on Web Services (ECOWS'07)*, pp. 129–136, Halle, Germany, November 2007.

[32] V. Viegas, J. M. D. Pereira, and P. M. B. S. Girão, ".NET framework and web services: a profit combination to implement and enhance the IEEE 1451.1 standard," *IEEE Transactions on*

Instrumentation and Measurement, vol. 56, no. 6, pp. 2739–2747, 2007.

[33] E. Song and K. Lee, "Smart transducer web services based on the IEEE 1451.0 standard," in *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (IMTC'07)*, May 2007.

[34] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: a survey," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 2, pp. 52–73, 2009.

[35] F. Hu and X. Cao, "Security in wireless actor & sensor networks (WASN): towards a hierarchical re-keying design," in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05)*, pp. 528–533, April 2005.

[36] M. Shao, S. Zhu, W. Zhang, G. Cao, and Y. Yang, "PDCS: security and privacy support for data-centric sensor networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1023–1038, 2009.

[37] A. S. Tanenbaum, *Computer Networks*, Prentice Hall, Upper Saddle River, NJ, USA, 4th edition, 2003.

[38] "Working Draft of ISO/IEC Draft Standard for Telecommunications and Information Exhange between Systems—Security framework for ubiquitous sensor network," ISO/IEC Unapproved Draft Std ISO/IEC, 29180, May 2008, http://isotc.iso.org/livelink/livelink?func=ll&objId=8158657&objAction=Open&vernum=1.

[39] M. MacDonald, *Microsoft .NET, Distributed Applications: Integrating XML Web Services and .NET Remoting*, Microsoft, Redmond, Wash, USA, 2003.

[40] W3 Consortium, XML Encryption specification, http://www.w3.org/.

[41] W3 Consortium, XML Signature specification, http://www.w3.org/TR/xmldsig-core/.

[42] A. Nadalin, C. Kaler, R. Monzillo, and P. Hallam-Baker, "Web services security: SOAP message security 1.1," OASIS Standard Specification, 2006.

[43] S. A. Kiprushkin, N. A. Korolev, and S. Y. Kurskov, "Distributed information measurement and control system for research and education in physics," in *Proceedings of the 2nd International Conference on Systems (ICONS'07)*, April 2007.

[44] J. Hieb, J. Graham, and S. Patel, "Security enhancements for distributed control systems," in *Critical Infrastructure Protection*, E. Goetz and S. Shenoi, Eds., pp. 133–146, Springer, Boston, Mass, USA, 2007.

[45] A. Aiello, D. L. Carnì, D. Grimaldi, and G. Guglielmelli, "Wireless distributed measurement system by using mobile devices," in *Proceedings of the 3rd IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'05)*, pp. 316–319, Sofia, Bulgaria, September 2005.

[46] A. Aiello, D. L. Carnì, D. Grimaldi, G. Guglielmelli, and F. Lamonaca, "Wireless distributed measurement system based on PDA and dynamical application repository server," in *Proceedings of the IEEE Instrumentation and Measurement Technology (IMTC'07)*, May 2007.

[47] L. Tao, H. Xu, and Z. Zhang, "Distributed inspecting and control system for motor vehicle safety performance," in *Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC'09)*, pp. 384–387, Zhejiang, China, August 2009.

[48] H. R. Tseng, R. H. Jan, and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks," in *Proceedings of the 50th Annual IEEE Global Telecommunications Conference (GLOBECOM'07)*, pp. 986–990, Washington, DC, USA, November 2007.

[49] D. Panda, "An Introduction to Service-Oriented Architecture from a Java Developer Perspective," http://www.onjava.com/pub/a/onjava/2005/01/26/soa-intro.html.

[50] N. Gruschka and N. Luttenberger, "Protecting web services from DoS attacks by SOAP message validation," in *Proceedings of the IFIP TC11 21 International Information Security Conference (SEC'06)*, May 2006.

[51] V. Shnayder, M. Hempstead, B. R. Chen, G. W. Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys'04)*, pp. 188–200, November 2004.