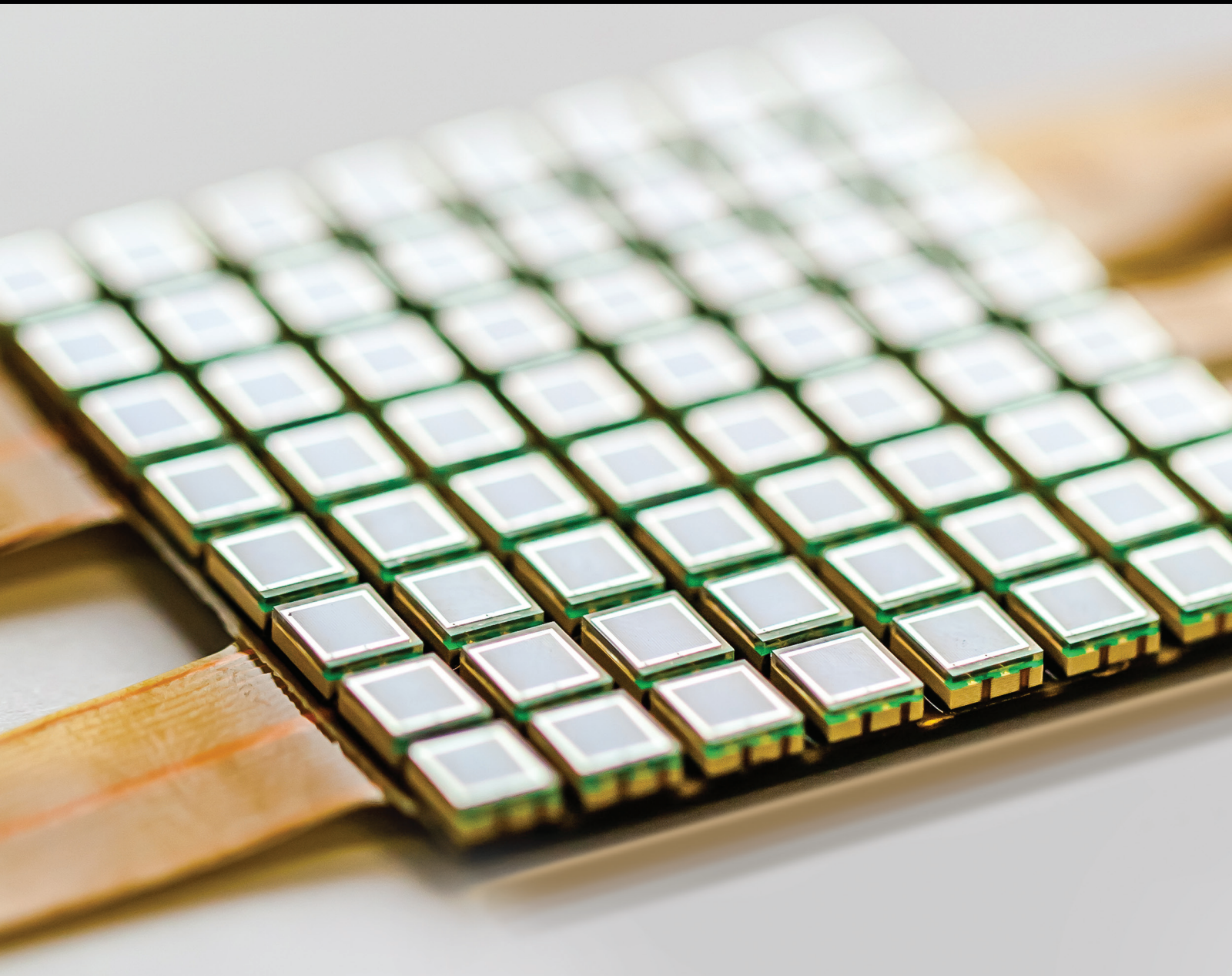


Recent Advances in Security and Privacy for Wireless Sensor Networks

Guest Editors: Fei Yu, Chin-Chen Chang, Jian Shu, Iftikhar Ahmad, Jun Zhang, and Jose Maria de Fuentes





Recent Advances in Security and Privacy for Wireless Sensor Networks

Recent Advances in Security and Privacy for Wireless Sensor Networks

Guest Editors: Fei Yu, Chin-Chen Chang, Jian Shu,
Iftikhar Ahmad, Jun Zhang, and Jose Maria de Fuentes



Copyright © 2015 Hindawi Publishing Corporation. All rights reserved.

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

Editorial Board

Harith Ahmad, Malaysia
Sheikh Akbar, USA
F. J. Arregui, Spain
Francesco Baldini, Italy
Romeo Bernini, Italy
Shekhar Bhansali, USA
Wojtek J. Bock, Canada
Hubert Brändle, Switzerland
Stefania Campopiano, Italy
Jian-Nong Cao, Hong Kong
Chi Chiu Chan, Singapore
Nick Chaniotakis, Greece
Nicola Cioffi, Italy
Elisabetta Comini, Italy
Marco Consales, Italy
Jesus Corres, Spain
Andrea Cusano, Italy
Dzung V. Dao, Japan
Manel del Valle, Spain
Ignacio Del Villar, Spain
Utkan Demirci, USA
Junhang Dong, USA
Omar Elmazria, France
Abdelhamid Errachid, France
Stephane Evoy, Canada
Xiao-Miao Feng, China
Vittorio Ferrari, Italy
Luca Francioso, Italy
Laurent Francis, Belgium
Paddy French, The Netherlands
Lung-Ming Fu, Taiwan
Mohammad Reza Ganjali, Iran
Wei Gao, Japan
Michele Giordano, Italy
K. Vengatajalabathy Gobi, India
Marco Grassi, Italy
Banshi D. Gupta, India
María del Carmen Horrillo, Spain
Wieslaw Jakubik, Poland
Hai-Feng Ji, USA
Kourosh Kalantar-Zadeh, Australia
Sher Bahadar Khan, Saudi Arabia
Sang Sub Kim, Republic of Korea
Won-Gun Koh, Korea
Challa Kumar, USA
Hiroki Kuwano, Japan
Laura M. Lechuga, Spain
Chengkuo Lee, Singapore
Jong-Jae Lee, Korea
Chenzhong Li, USA
Eduard Llobet, Spain
Yu-Lung Lo, Taiwan
Oleg Lupan, Moldova
Eugenio Martinelli, Italy
Yasuko Y. Maruo, Japan
Ignacio R. Matias, Spain
Mike McShane, USA
Igor L. Medintz, USA
Fanli Meng, China
Aldo Minardo, Italy
Joan Ramon Morante, Spain
Lucia Mosiello, Italy
Masayuki Nakamura, Japan
Liviu Nicu, France
Marimuthu Palaniswami, Australia
Gyuhae Park, Korea
Alain Pauly, France
Michele Penza, Italy
Andrea Ponzoni, Italy
Biswajeet Pradhan, Malaysia
Zhi-Mei Qi, China
Ioannis Raptis, Greece
Leonhard Reindl, Germany
Christos Riziotis, Greece
Maria Luz Rodríguez-Méndez, Spain
Albert Romano-Rodríguez, Spain
Josep Samitier, Spain
Giorgio Sberveglieri, Italy
Luca Schenato, Italy
Michael J. Schöning, Germany
Andreas Schütze, Germany
Woosuck Shin, Japan
Pietro Siciliano, Italy
Weilian Su, USA
Tong Sun, UK
Hidekuni Takao, Japan
Isao Takayanagi, Japan
Pierre Temple-Boyer, France
Guiyun Tian, UK
Suna Timur, Turkey
Jianhua Tong, China
Yu Chen Tsai, Taiwan
Hana Vaisocherova, Czech Republic
Joel Villatoro, Iran
Dong-ning Wang, Hong Kong
Qihao Weng, USA
Stanley E. Woodard, USA
Hai Xiao, USA
Jerliang Andrew Yeh, Taiwan
Hyeonseok Yoon, Korea
Wentao Zhang, China
Tao Zhu, China

Contents

Recent Advances in Security and Privacy for Wireless Sensor Networks, Fei Yu, Chin-Chen Chang, Jian Shu, Iftikhar Ahmad, Jun Zhang, and Jose Maria de Fuentes
Volume 2015, Article ID 169305, 2 pages

A Review on Sensor Network Issues and Robotics, Ji Hyoung Ryu, Muhammad Irfan, and Aamir Reyaz
Volume 2015, Article ID 140217, 14 pages

A Self-Adaptive Wireless Sensor Network Coverage Method for Intrusion Tolerance Based on Trust Value, Zuo Chen, Xue Li, Bing Yang, and Qian Zhang
Volume 2015, Article ID 430456, 10 pages

Accurately Identifying New QoS Violation Driven by High-Distributed Low-Rate Denial of Service Attacks Based on Multiple Observed Features, Jian Kang, Mei Yang, and Junyao Zhang
Volume 2015, Article ID 465402, 11 pages

Survey of Security Technologies on Wireless Sensor Networks, Qiuwei Yang, Xiaogang Zhu, Hongjuan Fu, and Xiqiang Che
Volume 2015, Article ID 842392, 9 pages

Research on Handoff Delay and Mobility Management Cost of Mobility Protocols in Wireless Sensor Networks, A. Q. Zhao and Y. Hu
Volume 2015, Article ID 179520, 8 pages

Trust-Aware and Low Energy Consumption Security Topology Protocol of Wireless Sensor Network, Zuo Chen, Min He, Wei Liang, and Kai Chen
Volume 2015, Article ID 716468, 10 pages

Sequence Alignment with Dynamic Divisor Generation for Keystroke Dynamics Based User Authentication, Jiacang Ho and Dae-Ki Kang
Volume 2015, Article ID 935986, 14 pages


Distributed Software-Attestation Defense against Sensor Worm Propagation, Jun-Won Ho
Volume 2015, Article ID 874782, 6 pages

B-iTRS: A Bio-Inspired Trusted Routing Scheme for Wireless Sensor Networks, Mingchuan Zhang, Ruijuan Zheng, Qingtao Wu, Wangyang Wei, Xiuling Bai, and Haixia Zhao
Volume 2015, Article ID 156843, 8 pages

A Novel Digital Certificate Based Remote Data Access Control Scheme in WSN, Wei Liang, Zhiqiang Ruan, Hongbo Zhou, and Yong Xie
Volume 2015, Article ID 928174, 11 pages

Parking Query in Vehicular Delay-Tolerant Networks with Privacy Protection Based on Secure Multiparty Computation, Haiping Huang, Juan Feng, Dan Sha, Jia Xu, and Hua Dai
Volume 2015, Article ID 420912, 8 pages

Sensor Networks Hierarchical Optimization Model for Security Monitoring in High-Speed Railway Transport Hub, Zhengyu Xie and Yong Qin
Volume 2015, Article ID 951242, 9 pages



Intrusion Detection Algorithm for Mitigating Sinkhole Attack on LEACH Protocol in Wireless Sensor Networks, Ranjeeth Kumar Sundararajan and Umamakeswari Arumugam
Volume 2015, Article ID 203814, 12 pages

A Data Processing Middleware Based on SOA for the Internet of Things, Feng Wang, Liang Hu, Jin Zhou, and Kuo Zhao
Volume 2015, Article ID 827045, 8 pages

Editorial

Recent Advances in Security and Privacy for Wireless Sensor Networks

**Fei Yu,¹ Chin-Chen Chang,² Jian Shu,³ Iftikhar Ahmad,⁴
Jun Zhang,⁵ and Jose Maria de Fuentes⁶**

¹*Peoples' Friendship University of Russia, Moscow 117198, Russia*

²*Feng Chia University, Taichung 40724, Taiwan*

³*Nanchang Hangkong University, Nanchang 330000, China*

⁴*King Saud University, Riyadh 92144, Saudi Arabia*

⁵*Deakin University, Burwood, VIC 3125, Australia*

⁶*Universidad Carlos III de Madrid, Madrid 28036, Spain*

Correspondence should be addressed to Fei Yu; hunanyufei@126.com

Received 20 May 2015; Accepted 27 May 2015

Copyright © 2015 Fei Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless networks have experienced explosive growth during the last few years. Nowadays, there are a large variety of networks spanning from the well-known cellular networks to noninfrastructure wireless networks such as mobile ad hoc networks and sensor networks. Communication security is essential to the success of wireless sensor network applications, especially for those mission-critical applications working in unattended and even hostile environments. However, providing satisfactory security protection in wireless sensor networks has ever been a challenging task due to various network and resource constraints and malicious attacks.

In this special issue, we concentrate mainly on security and privacy as well as the emerging applications of wireless sensor network. It aims to bring together researchers and practitioners from wireless and sensor networking, security, cryptography, and distributed computing communities, with the goal of promoting discussions and collaborations. We are interested in novel research on all aspects of security in wireless sensor networks and tradeoff between security and performance such as QoS, dependability, and scalability. The special issue covers industrial issues/applications and academic research into security and privacy for wireless sensor networks.

This special issue includes a collection of 14 papers selected from 74 submissions to 9 countries or districts

(China, India, South Korea, Chennai, USA, Pakistan, Saudi Arabia, Malaysia, and Taiwan). All submitted papers followed the same standard (peer-reviewed by at least three independent reviewers) as applied to regular submissions.

In the paper entitled “Accurately Identifying New QoS Violation Driven by High-Distributed Low-Rate Denial of Service Attacks Based on Multiple Observed Features,” J. Kang et al. propose using multiple observed features of network traffic to identify new high-distributed low-rate Quality of Services (QoS) violation so that detection accuracy may be further improved.

In the paper entitled “A Self-Adaptive Wireless Sensor Network Coverage Method for Intrusion Tolerance Based on Trust Value” Z. Chen et al. propose a network coverage method for invasive tolerance based on trust value of nodes by combining the trust value model with the reliable coverage optimization.

The paper entitled “Trust-Aware and Low Energy Consumption Security Topology Protocol of Wireless Sensor Network” by Z. Chen et al. tries to take node trust into consideration when building a network topology, so as to ensure the security of the network communication. The TLES algorithm is based on the analysis of node behavior. It develops a variety of trust factors and then performs comprehensive analysis with direct information and recommended information.

In the paper entitled “A Data Processing Middleware Based on SOA for the Internet of Things” by F. Wang et al., based on characteristics of the architecture and challenges of information fusion in the IoT, the paper designs a middleware platform based on SOA architecture for the integration of multisource heterogeneous information.

In the paper entitled “Survey of Security Technologies on Wireless Sensor Networks” Q. Yang et al. summarized research progress of sensor network security issues in key management, authentication, and secure routing as three aspects, by analyzing and commenting on these results advantages and disadvantages, and pointed out the future direction of the hot research field.

The paper entitled “Intrusion Detection Algorithm for Mitigating Sinkhole Attack on LEACH Protocol in Wireless Sensor Networks” by R. K. Sundararajan and U. Arumugam proposes an Intrusion Detection System (IDS) mechanism to detect the intruder in the network which uses Low Energy Adaptive Clustering Hierarchy (LEACH) protocol for its routing operation.

In the paper entitled “Parking Query in Vehicular Delay-Tolerant Networks with Privacy Protection Based on Secure Multiparty Computation” H. Huang et al. design the routing protocol RPAD from the direction and distribution density of the vehicle and the base station is considered to be a reference.

In the paper entitled “A Novel Digital Certificate Based Remote Data Access Control Scheme in WSN” by W. Liang et al. a digital certificate based remote data access control scheme is proposed for safe authentication of accessor in wireless sensor network (WSN).

The paper entitled “Distributed Software-Attestation Defense against Sensor Worm Propagation” by J.-W. Ho proposed on-demand software-attestation based scheme to stop worm propagation in sensor network.

In the paper entitled “Sensor Networks Hierarchical Optimization Model for Security Monitoring in High-Speed Railway Transport Hub” Z. Xie and Y. Qin consider the sensor networks hierarchical optimization problem in high-speed railway transport hub (HRTTH).

The paper entitled “B-iTRS: A Bio-Inspired Trusted Routing Scheme for Wireless Sensor Networks” by M. Zhang et al. presents a novel bio-inspired trusted routing scheme (BiTRS) based on ant colony optimization (ACO) and *Physarum* autonomic optimization (PAO).

The paper entitled “Research on Handoff Delay and Mobility Management Cost of Mobility Protocols in Wireless Sensor Networks” by A. Q. Zhao and Y. Hu focused on the research of mobility management cost of mobility support protocols and made analysis and comparison of mobility management cost among various mobility support protocols.

The paper entitled “Sequence Alignment with Dynamic Divisor Generation for Keystroke Dynamics Based User Authentication” by J. Ho and D.-K. Kang proposed sequence alignment with dynamic divisor generation (SADD) for user authentication by using the keystroke dynamics.

Acknowledgments

In particular, we would like to acknowledge the program committee members of Seventh International Symposium on Information Processing (ISIP 2014). This issue contains most of the revised and expanded versions of the selected quality papers presented at the Seventh International Symposium on Information Processing (ISIP 2014). We wish to express our deepest thanks to the program committee members for their help in selecting papers for this issue and especially the referees of the extended versions of the selected papers for their thorough reviews under a tight time schedule. ISIP 2014 took place on October 25-26, 2014, in Changsha, China, and was cosponsored by Jiangxi University of Science and Technology, China; Peoples' Friendship University of Russia, Russia; South China University of Technology, China; Feng Chia University, Taiwan; Henan Polytechnic University, China; Nanchang Hangkong University, China; and Jiangxi University of Science and Technology, China. In closing, we would like to take this opportunity to thank the authors for the efforts they put in the preparation of the manuscripts and in keeping the deadlines set by editorial requirements. We hope that you will enjoy reading this special issue as much as we did putting it together.

Fei Yu
 Chin-Chen Chang
 Jian Shu
 Iftikhar Ahmad
 Jun Zhang
 Jose Maria de Fuentes

Review Article

A Review on Sensor Network Issues and Robotics

Ji Hyoung Ryu,¹ Muhammad Irfan,² and Aamir Reyaz¹

¹Chonbuk National University, Jeonju, Republic of Korea

²The Infrastructure University Kuala Lumpur (IUKL), Kuala Lumpur, Malaysia

Correspondence should be addressed to Aamir Reyaz; aamir8095@gmail.com

Received 21 November 2014; Accepted 11 February 2015

Academic Editor: Iftikhar Ahmad

Copyright © 2015 Ji Hyoung Ryu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The interaction of distributed robotics and wireless sensor networks has led to the creation of mobile sensor networks. There has been an increasing interest in building mobile sensor networks and they are the favored class of WSNs in which mobility plays a key role in the execution of an application. More and more researches focus on development of mobile wireless sensor networks (MWSNs) due to its favorable advantages and applications. In WSNs robotics can play a crucial role, and integrating static nodes with mobile robots enhances the capabilities of both types of devices and enables new applications. In this paper we present an overview on mobile sensor networks in robotics and vice versa and robotic sensor network applications.

1. Introduction

Technological advances as well as the advent of 4G communications and of pervasive and ubiquitous computing have promoted a new interest in multihop networks (*ad hoc* communications). In particular, the interest is in self-organizing wireless multihop networks composed of a possibly large number of motes which can be mobile and static and can also be used for computational and power capabilities. Wireless sensor networks (WSNs) are typical examples of these kinds of networks. Most of the research in WSNs concerns networks whose nodes cannot be replaced and do not move. Mobility of the sensor nodes has been exploited for improving, or enabling altogether, communication coverage and sensing [1]. The credit for the creation of mobile sensor networks goes to wireless sensor networks and to the interaction of distributed robotics. The class of networks where small sensing devices in a collaborative way move in a space to observe and monitor environmental and physical conditions are known as mobile sensor networks [2]. Mobile sensor network is composed of nodes and all nodes have sensing, computation, communication, and locomotion modules (Figure 1). Each sensor node is capable of navigating autonomously or under the control of humans [3]. MSNs have emerged as an important area for research and development.

Though MSNs are still in the developing stages, they can be used for monitoring of environmental habitat, healthcare, agriculture, defense applications, disaster prone areas, hazardous zones, and so forth. MWSNs can also be used for monitored control, and more and more practical applications of MSNs also continue to emerge [4]. And robotics is the science of technology with applications in various fields such as design, fabrication, and theory [5]. It can also be considered as the area of technology dealing with construction, operation, control of robotic applications and computer systems, sensory feedback, and information processing. The main advantage of this technology is that it can replace humans in manufacturing processes and dangerous environments or can also resemble humans in behavior, cognition, or experience [5]. A breakthrough in the autonomous robot technology occurred in the mid-1980s with work in behavior based robotics. We can say that this work was the foundation for many current robotic applications [6]. By incorporating intelligent, mobile robots directly into sensor networks most of the problems in traditional sensor networks may be addressed. Mobile robots offer the ways to interact and survey the environment in a decentralized and dynamic way. The new system of robots and networked sensors led to the development of new solutions to the existing problems such as navigation and localization [7]. Mobile

nodes can be implemented as autonomous perceptive mobile robots or as perceptive robots whose sensor systems address environmental and navigational tasks. So, we can say robotic sensor networks are the distributed systems in which mobile robots carry sensors around an area to sense phenomena and to produce detailed environmental assessments [8]. The use of multirobot systems for carrying sensors around the environment represents a solution that has received a significant attention and can also provide some extraordinary advantages. A number of applications have been addressed so far by robotic sensor networks, such as rescue, search, and environmental monitoring. In wireless sensor networks, robotics can also be used to solve many problems to advance performances, such as responding to a particular sensor failure, node distribution, and data aggregation. Similarly for solving the problems that exist in the field of robotics wireless sensor networks can play a crucial role. Problems like localization, path planning, coordination for multiple robot, and sensing can be solved by using wireless sensor networks [9]. Today we have many applications of sensor networks on ground, air, underwater, and underground. In mobile UWSN sensor mobility can bring two major benefits. Floating sensors can increase system reusability and can also help to enable dynamic monitoring and coverage. Mobile sensors can help to track changes in water masses thus providing 4D (space and time) environmental monitoring. As compared to ground based sensor networks mobile UWSNs have to employ acoustic communications because in hard water environments radio does not work. Similarly underground sensor network can be used to monitor a variety of conditions such as properties of soil and environmental monitoring for toxic substances. They are buried completely underground and do not require any wired connections. On the ground they can be used for target tracking, environmental monitoring, detecting forest fire, industrial monitoring, and machine health monitoring. Wireless sensor nodes are into the service from a long time and were being used for different applications such as earthquake measurements and warfare. The recent growth of small sensor nodes dates back to the year 1998 NASA Sensor Webs project and smart dust project. To make autonomous sensing and communication possible within a cubic millimeter of space was the main purpose of the smart dust project. This project led to many more research projects including major research centers in CENS and Berkeley NEST. The term *mote* was coined by researchers working in these projects to refer to a sensor node; *pod* is the same term used in the NASA Sensor Webs project for a physical sensor node, although in a Sensor Web the sensor node can be another Sensor Web itself [10].

The main components of a sensor node are as follows: transceiver, a microcontroller external memory, one or more sensors, and power source [10]. The controller processes the data and controls functionality of other components in the sensor nodes. The feasible option of wireless transmission media is infrared, radio frequency (RF), and optical communication. As far as external memory is concerned the most relevant kinds of memory are the flash memory and the on-chip memory of a microcontroller. The most important feature in the development of a wireless sensor node is to

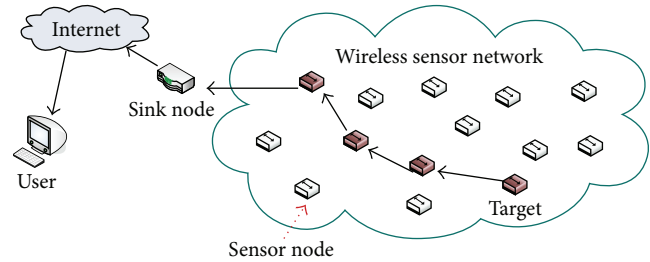


FIGURE 1: Wireless sensor network architecture.



FIGURE 2: Temperature humidity sensor module.

make sure that there is always sufficient energy available to the power system. Sensor nodes consume power for data processing, sensing, and communication; power is stored in capacitors or batteries. Batteries can be both rechargeable and nonrechargeable and for sensor nodes they are the main resource of power supply. And, sensor is a device that detects or senses heat, light, sound, motion, and so forth and then responds to it in a particular way [11] (Figure 2).

The crossbow radio/processor boards usually known as motes permit many sensors scattered over a large area to wirelessly transmit their data back to the base station which is attached to the computer (Figure 3). These motes run TinyOS operating system which is an open source operating system designed for low-power wireless devices, such as those used in PANs, smart meters, ubiquitous computing, sensor networks, and smart buildings [12]. It controls power, radio transmission and networking transparent to the user, and the network which is formed as an *ad hoc* network [13].

The MICA2 Mote is a third generation mote module with 512 Kbytes of measurement (serial) flash memory, 128 Kbytes of program flash memory, and 4 Kbytes of programmable read-only memory (Figure 4).

Stargate is a 400 MHz Intel PXA255 Xscale processor with 32 Mbytes of flash memory and 64 Mbytes of synchronous dynamic random access memory. A number of classes of sensors are available; these include barometric pressure, acceleration, seismic, acoustic, radar, magnetic camera, light, temperature, relative humidity, magnetic camera, and global positioning system (GPS). Usually sensors are classified into 3 types: passive, omnidirectional, passive, narrow-beam, and active sensors. Passive sensors are self-powered; they sense the data without actually manipulating the environment by active probing while active sensors actively probe the environment. Narrow beam sensors have a well-defined notion of direction of measurement. Omnidirectional sensors have no notion of direction involved in their measurements [10] (Figure 5).

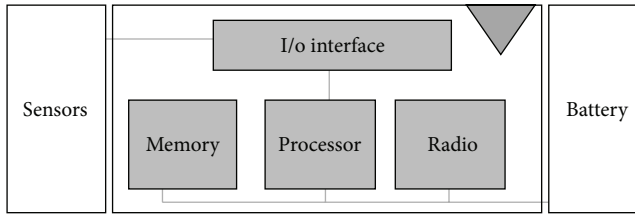


FIGURE 3: A sensor node architecture.



FIGURE 4: MICA2 processor.

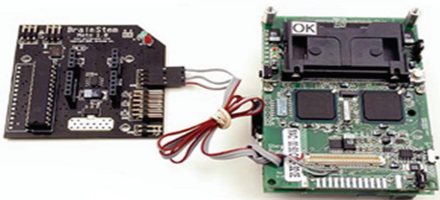


FIGURE 5: Stargate processor.

2. Mobile Sensor Networks

Mobile sensor networks are a class of networks where small sensing devices move in a space over time to collaboratively monitor physical and environmental conditions [2]. The research on mobile sensor networks has been plenty worldwide. For MSN, there could be a lot of valuable application with attached sensors as well as capabilities such as locomotion, environmental information sensing, and dead reckoning. The architecture of MSNs can be divided into node, server, and client layer [3]. The job of the node layer is to acquire all sorts of data, as it is directly embedded into physical world. This layer also consists of all the static as well as mobile sensor nodes. Server layer comprises single board computer running server software or a personal computer. The client layer devices can be any smart terminals, these devices also include remote and local clients. Mobility is an unrealistic or undesirable characteristic of sensor nodes as it can address the objective challenges [14]. Research issues on Mobile sensor networks can be analyzed into two aspects [2]: communication issues and data management issues. Our work is focused on communication issues which include coverage and localization issues (Figure 6).

2.1. Coverage. In the sensor networks, coverage can be seen as the measure of quality of service. The quality of surveillance that the network can provide also depends upon the coverage

of a sensor network [15, 16]. It can be seen that, for all the applications of mobile sensor networks, coverage is one of the most fundamental issues [17]. It will decrease due to sensor failure and undesirable sensor deployment. Gage (Gage 92) defines coverage as the maintenance of spatial relationship which adjusts to exact local conditions to optimize the performances of some functions. Gage describes three coverage behavior types. *Blanket coverage*: its objective is to achieve a static arrangement of nodes that minimizes the total detection area. *Barrier coverage*: the main goal of the barrier coverage is to reduce the probability of unnoticed penetration through the barrier. *Sweep coverage*: the concept of sweep coverage is from robotics which is more or less equivalent to moving barrier. The lifetime of sensors is strongly affected by hardware defects, battery depletions, some harsh external environments (e.g., fire, wind), and so forth [2]. In MSNs, previously uncovered areas became covered when sensors move through them and when sensors move away, the already covered areas become uncovered. As a result, the areas covered by sensors change over time, and more areas will be covered at least once as time continues. For robotic applications Khatib [18] was the first one to describe potential field techniques for tasks like local navigation and obstacle avoidance. Similar concept of “motor schemas” was also introduced which uses the superposition of spatial vectors to generate behavior [19]. Howard et al. [20] also used potential fields, but for the deployment problem, they consider the problem of arranging mobile sensors in an unknown environment, where fields are constructed such that each node is repelled by other nodes and also throughout the environment obstacles forces the network to spread. Reference [21] also proposed potential field technique which is distributed and scalable and does not require prior map of the environment. In [22], for the uncovered areas by the sensor network, new nodes are always placed on the boundary of those areas. It is also able to find a suboptimal deployment solution and also makes it sure that each node must be in line of sight with another node. In order to increase the coverage [23] proposed algorithms to calculate the desired target positions where sensors should move and identify the coverage holes existing in the network. To find out the coverage holes Voronoi diagram was used by Wang et al. [24] and he also designed three movement-assisted sensor deployment protocols, namely, VEC (vector based), VOR (Voronoi-based), and Minimax, based on the principle of sensors moving from densely deployed areas to sparsely deployed areas. A virtual force algorithm (VFA) was proposed by [25] to increase sensor field coverage by combining repulsive forces and attractive forces to determine randomly deployed sensors movement and virtual motion paths. Reference [26] deals with both static and mobile sensors and within the sensor field the job is to be served by mobile sensors which appear at random locations. The static sensors then guide the mobile sensors to the position where the task occurs when they get aware about the arrival of tasks. Researchers deal with the dynamic aspects of coverage in mobile sensor networks and also characterized area coverage at specific time instants and during time interval and detection time of the randomly located target [27]. The problem of coverage and exploration

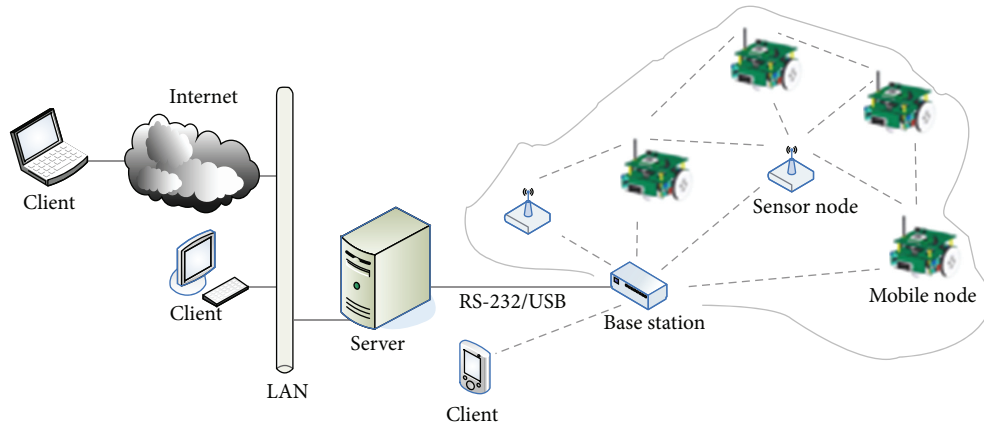


FIGURE 6: The system architecture of a mobile sensor network.

through the utilization of deployed network was considered and the algorithm which assumes that the global information is not available was also presented [28]. In [29] the problem of sensor relocation is being focused and two-phase sensor relocation solution has been proposed in which redundant sensors are identified first using Grid-Quorum and then are relocated in a cascaded movement in a timely, efficient, and balanced way [2].

2.2. Localization. Recently, there has been much focus on building mobile sensors, and we have seen the development of small-profile sensing devices that are quite capable of controlling their own movement. Mobility has become an important area of research for mobile sensor networks. Mobility enables sensor nodes to target and track moving phenomena such as vehicles, chemical clouds, and packages [30]. One of the most significant challenges for mobile sensor nodes is the need for localization. Localization is the ability of sensor nodes to find out its physical coordinates, and localization on mobile sensors is performed for navigational and tracking purposes. Localization is required in many applications in wireless sensor networks such as health, military, and industry. Extensive research has been done so far on localization. Many location discovery schemes have proposed to eliminate the need of GPS on every sensor node [31]. GPS is commonly considered to be a good solution for outdoor localization. However, GPS is still expensive and hence insufficient to be used for large number of devices in WSN. Some of the problems with GPS are as follows.

These are some situations in which GPS will not work reliably because GPS receiver needs line of sight to the multiple satellites and it does not work well in the indoor environment. And GPS receivers are available for mote scale devices only. They are still expensive and undesirable for many applications. Even if GPS receivers became cheaper and are used in every node, the nodes cannot actively use GPS in mobile sensor networks. Typically GPS node consumes more energy than sensors and low-power transceivers. The problem of using GPS in a real environment also exists in GPS itself. GPS shows 10~20 m of error when used in normal outdoor environments unless it uses a costly mechanism such as

differential GPS. Deploying a large number of GPS in mobile sensor network has both limits and possibilities [32]. There are two types of localization algorithms, namely, centralized and distributed algorithms [31]. These centralized location techniques depend on sensor nodes transmitting data to a central location, where computation is performed to find out the location of each node [33]. Distributed algorithms do not need a central base station and for determining its location it relies on each node with only limited communication with nearby nodes [31]. Localization algorithms in MWSNs can be categorized into (1) range-based method, (2) range-free method, (3) mobility based method [2]. All of these methods vary in the information used for localization purposes. Range-based methods use range measurements while range-free techniques only use the content of messages [34]. Range-based methods also require expensive hardware to measure signal arrival time and angle of signal arrival. As compared to range-free methods these methods are expensive because of their pricey hardware [2]. Range-based approaches have also utilized time of arrival, received signal strength, time difference of arrival of two different signals (TDOA), and the angle of arrival. Though they can reach the fine resolution, either the required hardware is expensive or the results depend on impractical assumptions about signal propagation [33]. While range-free methods use local and hop count techniques, for range-based approaches these methods are very cost effective. Many localization algorithms have been proposed so far such as elastic localization algorithm (ELA) and mobile geographic distributed localization algorithm; both of these algorithms assume unlimited storage in sensor nodes [2]. For sensor networks [33] two types of range-free algorithms have been proposed: local techniques and hop count techniques. *Local techniques* rely on high speed on a high density of seeds so that every node can hear several seeds and hop count techniques rely on a flooding network. Each node in centroid method estimates its location by calculating the center of the seeds locations it hears. Location error can be reduced if seeds are well positioned but in *ad hoc* deployments this is impossible. The APIT method separates the environment into triangular regions between beaconing nodes and to calculate the maximum area it uses the grid

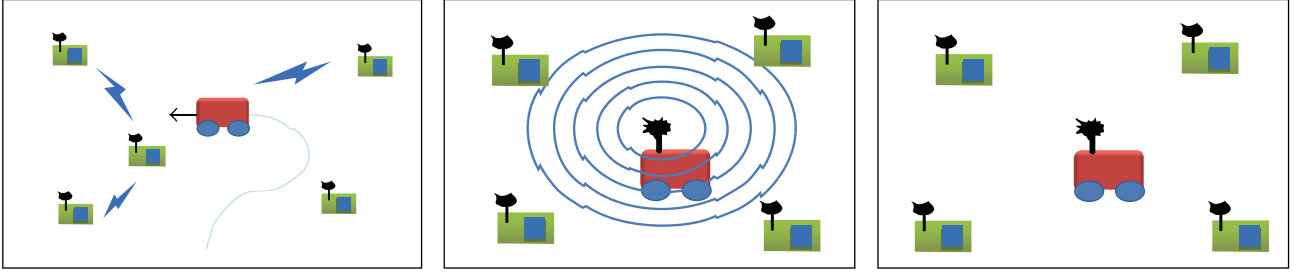


FIGURE 7: Coordination, measurement, and location estimation phase.

algorithm in which a node will likely reside [33]. *Hop count techniques* propagate the location estimation throughout the network where the seed density is low. In *mobility-based methods*, to improve accuracy and precision of localization method sequential Monte Carlo localization (SML) was proposed [27] without additional hardware except for GPS [2]. And without decreasing the nonlimited computational ability many techniques using SML are also being proposed. In order to achieve the accurate localization researchers proposed many algorithms, using the principles of Doppler shift and radio interferometry to achieve the accurate localization has also been used [2]. The three phases typically used in localization are (1) Coordination, (2) measurement, and (3) position estimation [30] (Figure 7).

To initiate the localization a group of nodes coordinate first, a signal is then emitted by some nodes, and then some property of the signal is observed by some other nodes. By transforming the signal measurements into position estimates node position is then determined. To find the positions of sensors in order to reduce the frequency of localization, three techniques were proposed: static fixed rate (SFR), dynamic velocity monotonic (DVM) and mobility aware dead reckoning driven (MADRD) [35].

- (1) Static fixed rate (SFR): the performance of this protocol varies with the mobility of sensors. In this base protocol each sensor invokes its localization periodically with a fixed time period $t_s f_r$. In this technique, the error will be high if the sensor is moving quickly and if it is moving slowly the error will be low [36]:

$$e_{sfr} = \frac{n \times \sin \theta}{\sin \alpha}. \quad (1)$$

- (2) Dynamic velocity monotonic: this is an adaptive protocol; with the mobility of sensors localization is called adaptively in DVM; the higher the observed velocity is, the faster the node should localize to maintain the same level of error. It computes the node velocity when it localizes by dividing the distance it has moved since the last localization point by the time that elapsed since localization. The next localization point is scheduled based on the velocity at the time when a prespecified distance will be travelled if the node continues with the same velocity [37].

- (3) Mobility aware dead reckoning driven (MADRD): to predict the future mobility this protocol computes the mobility pattern of the sensors. When the expected difference between the predicted mobility and expected mobility reaches the error threshold the localization should be triggered [38]:

$$\text{Err}_{\text{mardr}} = \int_0^T E \left[(x - x')^2 + (y + y')^2 \right] dt. \quad (2)$$

Mobility aware interpolation (MAINT) was proposed to estimate the current position with better trade-off between energy consumption and accuracy [37]. Their method uses interpolation which gives better estimation in most cases:

$$\text{Err}_{\text{maint}} = \int_0^T E \left[(x - x'')^2 + (y + y'')^2 \right] dt. \quad (3)$$

2.3. Positioning Systems. Many methods have been used for the problem of localization. Positioning systems will use positioning technology to determine the position and orientation of an object or person in a room [39].

2.3.1. Xbee Technology. It is a brand of radios that support a variety of communication protocols. It uses Zigbee protocol; Zigbee is a wireless communication protocol like Wi-Fi and Bluetooth. These modules use the IEEE 802.15.4 networking protocol for fast point-to-multipoint or peer-to-peer networking. Its low-power consumption limits transmission distances to 10–100-meter line of sight though, depending on power output and environmental characteristics [40]. It operates in unlicensed ISM bands, so it is prone to interference from a wide range of signal types using the same frequency which can disrupt the radio frequency. From RSSI values the distance between two Zigbee nodes is calculated; based on the distance calculated from Zigbee modules many researchers found it suitable for indoor localization. It comes with modules, though orientation is the problem in many modules because most of its modules are without omnidirectional antenna (Figure 8).

2.3.2. Wi-Fi Based Indoor Localization. Wi-Fi, or wireless networking, is one of the biggest changes to the way we use computers since the PC was introduced. Wi-Fi also allows communications directly from one computer to another



FIGURE 8: Xbee.

without an access point intermediary. This is called *ad hoc* Wi-Fi transmission. A typical wireless access point using 802.11b or 802.11g with a stock antenna might have a range of 35 m (115 ft) indoors and 100 m (330 ft) outdoors [41]. The widespread availability of wireless networks (Wi-Fi) has created an increased interest in harnessing them for other purposes, such as localizing mobile devices. There has long been interest in the ability to determine the physical location of a device given only Wi-Fi signal strength. This problem is called Wi-Fi localization and has important applications in activity recognition, robotics, and surveillance. The key challenge of localization is overcoming the unpredictability of Wi-Fi signal propagation through indoor environments. The data distribution may vary based on changes in temperature and humidity, as well as position of moving obstacles, such as people walking throughout the building. The uncertainty makes it difficult to generate accurate estimates of signal strength measurements. Wi-Fi based systems have number of issues, including high power consumption, being limited to coverage, and being prone to interference [42].

2.3.3. Ultrawideband and FM Radio Based Technique. To achieve high bandwidth connections with low-power consumption ultrawideband is the best communication method. Ultrawideband wireless radios send short signal pulses over broad spectrum [43]. This technology has been used in a variety of localization tasks requiring higher accuracy 20–30 cm than achievable through conventional wireless technologies, for example, radio frequency identification (RFID), WLAN, and so forth [41]. The limitation with this technique is that it requires specialized hardware and dedicated infrastructure, resulting in high costs for wide adoption [42]. FM radios can be used for indoor localization, while providing longer battery life than Wi-Fi, making FM an alternative to consider for positioning. FM radio signals are less affected by weather

conditions such as rain and fog in comparison to Wi-Fi or GSM. These signals penetrate walls easily as compared to Wi-Fi; this makes sure high availability of FM positioning signals in indoor environments. FM radio uses frequency division multiple access (FDMA) approach which splits the band into number of frequency channels that are used by stations. There are only few papers dedicated to FM radio based positioning.

2.3.4. Bluetooth Positioning System. By executing the inquiry protocol Bluetooth device detects other devices. Devices within its range that are set to “discoverable” will respond by identifying themselves. Bluetooth communicates using radio waves with frequencies between 2.402 GHz and 2.480 GHz, which is within the 2.4 GHz ISM frequency band, a frequency band that has been set aside for industrial, scientific, and medical devices by international agreement. The main advantage of using Bluetooth is that this technology is of high security, low power, low cost, and small size. Many researchers have used Bluetooth for indoor positioning and this reference used Bluetooth [44].

2.3.5. Radio Frequency Identification. An RFID (radio frequency identification) system consists of a reader with an antenna which interrogates nearby active transceivers or passive tags. Using RFID technology data can be transmitted from RFID tags to the reader via radio waves. The data consists of the tags unique ID (i.e., its serial number) which can be related to available position information of the RFID tag. These are used in localization because of their advantages; radio waves can pass through walls, obstacles, and human bodies easily. This technique needs less hardware and has large coverage area. By using advanced identification technology and noncontact, this technology uses one way one wireless communication that uses radio signals to put an RFID tag on objects and people to track them and automatically identify them [41].

2.3.6. Hybrid Positioning System. These systems use several different technologies to find the location of a mobile device by using many positioning technologies. To overcome the limitations of GPS, these systems are mainly developed because GPS does not work well in indoors. This system is being highly investigated for commercial and civilian based location services like Google maps for mobile, devicescape, and so forth [41, 45].

2.3.7. Quick Response Code. It is a matrix code that keeps a comparative huge amount of location information compared to standard barcode. It can be attached to some key areas of the buildings to wait for scanning to provide its positional information from database [46].

In this section we present a brief overview of the localization methods used by some researchers. In localization using RSSI, based on Xbee modules in wireless sensor networks, [47] proposed a localization technique using RSSI, and the technique is based on decision tree obtained from a set of empirical experiments. By applying the Cramer’s rule approach they used the decision tree to select the best three

neighbor reference nodes that are involved in the estimation of the position of target sensor node. The results obtained based on empirical data and gathered from the experiments indicate accuracy less than 2 meters. By using Zigbee CC2431 modules [48] proposed closer tracking algorithm for indoor wireless sensor localization. The proposed algorithm can suitably select an adaptive mode to obtain precise locations. They also improved the fingerprinting algorithm in mean time. The proposed technique CTA can determine the position with error less than 1 meter. Based on artificial neural networks [49] presented location estimation system in the indoor environment. This architecture provides robust mechanism for coping with unavailable information in real life situations as they employ modular multilayer perceptron (MMLP) approach to effectively reduce the uncertainty in the location estimation system. Moreover their system does not require runtime searching of nearest neighbors in huge backend database. Yu proposed a measurement and simulation based work of a fingerprinting technique based on neural networks and ultrawideband signals. Their proposed technique is based on the construction of a fingerprinting database of LDPs extracted from an UWB measurement campaign. To learn the database and to locate the targeted positions the feed forward neural network with incremental back backpropagation is used. In order to evaluate the positioning performance, different types of fingerprinting database and different sizes are considered [50]. To perform indoor localization, [51] evaluated several ANN designs by exploiting RSS fingerprints collected in an office environment. They relied on WLAN infrastructure to minimize the deployment cost. The proposed cRBF algorithm can be a good solution to the location estimation problem in indoor environments. Moreover based on their experimental results, it is found that the proposed algorithm achieves more accuracy compared to sRBF, MLP and GRNN designs, and KNN algorithm. Reference [52] used RSS fingerprinting technique and artificial neural network for mobile station location in the indoor environment. The proposed system learns offline the location “signatures” from extracted location-dependent features of the measured data for LOS and NLOS situations and then it matches online the observation received from a mobile station against the learned set of “signatures” to accurately determine its position. It was found that location precision of the proposed system is 0.5 meters for 90% of trained data and 5 meters for 45% of untrained data. By using RSSI values of anchor node beacons, researchers presented an artificial feed forward neural network based approach for node localization in sensor networks. They evaluated five different training algorithms to obtain the algorithm that gives best results. The multilayer perceptron (MLP) neural network has been obtained using the Matlab software and implemented using the Arduino programming language on the mobile node to evaluate its performance in real time environment. By using 12-12-2 feed forward neural network structure, an average error of 30 cm is obtained [53]. To infer the clients position in the wireless local area network (LAN), researchers presented a novel localization algorithm, namely discriminant adaptive neural network (DANN), which takes, received signal strength (RSS) from access points (APs) as an input. For

network learning, they extracted the useful information into discriminative components (DCs). This approach incrementally inserts DCs and recursively updates the weightings in the network until no further improvement is required. Traditional approaches were implemented on the same test bed, including weighted-nearest neighbor (WKNN), maximum likelihood (ML), and multilayer perceptron (MLP), and then results were compared. The results showed that the proposed technique has better results as compared to other examined techniques [54]. Ali used neural networks to solve the problem of localization in sensor networks. They compared the performances of three different families of neural networks: multilayer perceptron (MLP), radial basis function (RBF), and recurrent neural networks (RNN). They compared these networks with two variants of Kalman filter which are also used for localization. Resource requirements in terms of computational and memory resources were also compared. The experimental results in [55] show that RBF neural network has the best performance in terms of accuracy and MLP neural networks has best computational and memory resource requirement. Another neural network based approach was used by Laslo. For the processing, received signal strength indicator (RSSI) was used and its also used for learning of neural network and preprocessed (mean, median and standard deviation) in order to increase the accuracy of the system. Fingerprint (FP) localization methodology was also applied in the indoor experimental environment which is also presented. The RSSI values used for the learning of the neural network are preprocessed (mean, median, and standard deviation) in order to increase the accuracy of the system. To determine the accuracy of the neural network, mean square error of Euclidean distance between calculated and real coordinates and the histogram was used in [56]. Wi-Fi based technique was proposed for detecting users position in an indoor environment. They implemented the trilateration technique for localization. They used mobile phone to obtain RSSI from the access points and then the RSSI data was converted into distance between users and each AP. They also proposed to determine the users position based on trilateration technique [57]:

$$\text{distance}, d_i = p(1 - m_i), \quad (4)$$

where m is the percentage of signal strength, p is the maximum coverage of signal strength, and $I = 1, 2, 3$. By analyzing a large number of experimental data, it is found that the variance of RSSI value changes along with the distance regularly. They proposed the relationship function of the variance of RSSI and distance and establish the log-normal shadowing model with dynamic variance LNSM-DV based on the result analysis. Results show that LNSM-DV can further reduce error and have strong self-adaptability to various environments compared with LNSM [58]. In this study the problem of indoor localization using wireless Ethernet IEEE 802.11 (Wireless Fidelity, Wi-Fi) was analyzed. The main purpose of this work was to examine several aspects of location fingerprinting based on indoor localization that affects positioning accuracy. The results showed that they achieved the accuracy of 2–2.5 meters [59]. In the mobile

node localization, a system was proposed for real environment when both anchors and unknown nodes are moving. To figure out the current position, history of anchor information was used. User's movement was modeled by the archived information and for discovering new positions movement models were also used. In complex situations where anchors and nodes are mobile, researchers presented three methods to resolve localization problem [38]. The proposed methods take into account the capability of nodes: nodes which can calculate either distances or angles with their neighbors or none of both. When the sensor has at least two anchors in the neighborhood, the proposed methods determine the exact position; else it gives a fairly accurate position and in this case it can compute the generated maximal error. The proposed method also defines periods when a node has to invoke its localization. A GPS free localization algorithm was in MWSNs [60]. In order to build the coordinate system, the proposed algorithm uses the distance between the nodes and also nodes positions are computed in two dimensions. Based on dead reckoning, Tilak et al. [36] proposed a number of techniques for tracking mobile sensors. Among all the proposed techniques, mobility aware dead reckoning estimates the position of the sensor, instead of localizing the sensor every time it moves. Error in the estimated position is calculated every time the localization is called and with the time error in the estimation grows. And also the next localization time is fixed depending on the value of this error. For a given level of accuracy in position estimation fast mobile sensors trigger localization with higher frequency. Instead of localizing sensor, a technique was proposed to estimate the positions of a mobile sensor [37]. It gives higher accuracy for particular energy cost and vice versa. When sensor has some data to be sent, the position of the sensor is required then only. The information of an inactive sensor is ceased to be communicated. In order to reduce the arithmetic complexity of sensors, most calculations are carried out at base station. To solve the set of equations, [31] proposed the novel algorithm for MSN localization in which they took three nodes which are neighbors to each other and if the solutions are unique they are the node positions. And, for searching the position of the final node, a scan algorithm was also introduced called a metric average localization error (ALE) (which is the root mean square error of node locations divided by the total number of nodes) to evaluate the localization error:

$$\text{ALE} = \frac{\sum_{i=1}^N \|p_i - K_i\|}{N}. \quad (5)$$

3. Security Issues in Mobile Ad Hoc Sensor Networks

Because of the vulnerability of wireless links, nodes limited physical protection, nonexistence of certification authority, and lack of management point or a centralized monitoring, it is difficult to achieve security in mobile *ad hoc* networks [61]. Sensor networks have many applications: they are used in ecological, military, and health related areas. These applications often examine some sensitive information such

as location detection or enemy movement on the battlefields. Therefore security is very important in WSN. WSN has many limitations such as small memory, limited energy resources, and use of insecure channels in communication; these problems make security in WSN a challenge [62]. Designing the security schemes of wireless sensor networks is not an easy task; sensor networks have many constraints compared to computer networks [63]. The main objective of the security service in WSN is to protect the valuable information and resources from misbehavior and attacks; requirements in wireless sensor network security include availability, authorization, confidentiality, authentication, integrity, nonrepudiation, and freshness. Attacks in WSN can be categorized as follows: *attacks on network availability*: these are often referred to as DOS (denial of service attacks); these attacks can target any layer of a sensor network. *Attacks on secrecy and authentication* include packet replay attacks, eavesdropping, or spoofing of attacks. There are also *stealthy attacks against service integrity*. In this type of attack the motive of the attacker is to make network accept false data value. Security threats and issues in WSN can be classified into two categories: active and passive attacks [63].

3.1. Active Attacks. It implies the disruption of the normal functionality of the network, meaning information interruption, modification, or fabrication [64]. Impersonating, modification, fabrication, message replay, and jamming are the examples of active attacks. The following attacks are active in nature.

3.2. Routing Attacks in Sensor Networks. Routing attacks are the attacks which act on the network layer. While routing the messages many attacks can happen; some of them are as follows: attacks on information in transit, selective forwarding, and Blackhole/Sinkhole attack.

3.3. Wormhole Attacks. In this type of attack, the attacker records packets at one location in the network and then tunnels them to another location and retransmits them there into the network [65].

3.4. HELLO Flood Attacks. In this type of attack the attacker uses HELLO packets to convince the sensors in WSN. The attacker sends routing protocols HELLO packets from one node to another with more energy [63].

3.5. Denial of Service (DOS). These attacks can target any layer of a sensor network. This type of service is produced by unintentional failure of nodes or malicious action.

3.6. Node Subversion. Capture of a node may disclose its information and thus compromise the whole sensor network. In this attack the information stored on sensor might be obtained by attacking it [66].

3.7. Node Outage. When a node stops its function, this kind of situation occurs. In the case where a cluster leader stops

functioning, the sensor network protocols should be robust enough to mitigate the effects of node outages by providing an alternate route.

3.8. Node Malfunction. It can expose the integrity of a sensor network if the node starts malfunctioning.

3.9. Physical Attacks. These types of attacks can destroy sensors permanently. These attacks on sensor networks typically operate in hostile outdoor environments. Attackers can extract cryptographic secrets, modify programming in the sensors, tamper with associated circuitry, and so forth.

3.10. False Node. This can lead to injection of malicious data by the additional node; this can spread to all the nodes, potentially destroying the whole network. In this case an intruder adds a node to the system that feeds false data or it can also prevent the passage of true data.

3.11. Message Corruption. Any change in the content of a message by an attacker compromises its integrity [63].

3.12. Node Replication Attacks. In this attack, attacker adds an additional node to the existing network by copying the node ID. This can result in a disconnected sensor network and false sensor readings.

3.13. Passive Information Gathering. Strong encryption techniques need to be used in order to minimize the threats of passive information gathering [63].

3.14. Passive Attacks. In passive attacks, the attack obtains data exchange in the network without interrupting the communication [64]. Some of the most common attacks against sensor privacy are as the following.

3.14.1. Monitoring and Eavesdropping. The most common attack to the privacy is the monitor and eavesdropping. Eavesdropping is the intercepting and reading of messages and conversations by unintended users [64].

3.14.2. Traffic Analysis. There is a high possibility analysis of communication patterns even when the messages transferred are encrypted. The activities of sensor can disclose a lot of information to enable an adversary to cause malicious harm to the network.

3.14.3. Camouflage Adversaries. In the network one can comprise the nodes to hide or can insert their node. After inserting their node, that node can work as a normal node to attract packets and then misroute them which can affect privacy analysis.

4. Robotic Sensor Network Applications

Most of the problems in traditional sensor networks may be addressed by incorporating intelligent, mobile robots directly

into it. Mobile robots provide the means to explore and interact with the environment in a dynamic and decentralized way. In addition to enabling mission capabilities well beyond those provided by sensor networks, these new systems of networked sensors and robots allow for the development of new solutions to classical problems such as localization and navigation [1]. Many problems in sensor networks can be solved when putting robotics into use, problems like node positioning and localization, acting as data mule, detecting and reacting to sensor failure, for nodes mobile battery chargers, and so forth. And also wireless sensor networks can help solve many problems in robotics such as robot path planning, localization mapping, and sensing in robots [9]. Mobile nodes can be implemented as autonomous perceptive mobile robots or as perceptive robots whose sensor systems address environmental task and navigational task. Robotic sensor networks are particular mobile sensor networks or we can say robotic sensor networks are distributed systems in which mobile robots carry sensors around an environment to sense phenomena and to produce in depth environmental assessments. There are many applications of wireless sensor networks in robotics like robotics advanced sensing, coordination in robots, robot path planning, and robot localization, robot navigation, network coverage, proper data communication, data collection, and so forth. Using WSN helps emergency response robots to be conscious of the conditions such as electromagnetic field monitoring and forest fire detection. These networks improve the sensing capability and can also help robot in finding the way to the area of interest. WSNs can be helpful for coordinating multiple robots and swarm robotics because the network can assist the swarm to share sensor data, tracking its members, and so forth. To perform the coordinated tasks it sends robots to the different locations and also a swarm takes decisions based on the localization of events, allowing path planning and coordination for multiple robots to happen efficiently and optimally and direct the swarm members to the area of interest. In the localization part there are many techniques for localizing robots within a sensor network. Cameras have been put into use to identify the sensors equipped with infrared light to triangulate themselves based on distances derived from pixel size. A modified SLAM algorithm has been utilized by some methods which uses robots to localize itself within the environment and then compensates for SLAM sensor error by fusing the estimated location with the estimated location in WSN based on RSSI triangulation [9]. An intruder detection system was presented in [59] which uses both wireless sensor networks and robots. In order to learn and detect intruders in previously unknown environment, sensor network uses an unsupervised fuzzy adaptive resonance theory (ART) neural network. A mobile robot travels upon the detection of an intruder to the position where the intruder is detected. The wireless sensor network uses a hierarchical communication/learning structure where mobile robot is root node of the tree.

In wireless sensor networks robotics can also play a crucial role. They can be used for replacing broken nodes, repositioning nodes, recharging batteries, and so forth. To increase the feasibility of WSNs, [67] used robots because

they have actuation but limited coverage in sensing while sensor networks lack actuation but they can acquire data. In servicing WSNs, robot task allocation and robot task fulfillment was examined [68]. Problems are examined in robot task allocation such as using multitask or single-task robots in a network and how to organize their behavior to optimally service the network. The route which a robot takes to service nodes is examined in robot task fulfillment. To improve the robot localization, [69] adapts sensor network models with information maps and then checks the capability of such maps to improve the localization. The node replacement application was developed by [70] in which a robot would navigate a sensor network based on RSSI (received signal strength indication) from nearby nodes and it would then send a help signal if a node will begin to run low on power. And, then through the network, the help signal would be passed to direct the robot to replace the node. Robots can also be used to recharge batteries. The problem of localization can also be solved using robots; they can be used to localize the nodes in the network. They can also be used in data aggregation. In a network, they can also serve as data mules; data mules are robots that move around the sensor network to collect data from the nodes and then transport that data back to sink node or perform the aggregation operation on data [9]. The use of multirobot systems for carrying sensors around the environment represents a solution that has received a considerable attention and can provide some remarkable advantages as well. A number of applications have been addressed so far by robotic sensor networks, including environmental monitoring and search and rescue. When put into use robotics sensor networks can be used for effective search and rescue, monitoring electromagnetic fields, and so forth. Search and rescue systems should quickly and accurately locate victims and map search space and locations of victims and with human responders it should maintain communication. For a search and rescue system to fulfill its mission, the system should be proficient to rapidly and reliably trace its victims within the search space and should also be well capable to handle a dynamic and potentially hostile environment. For utilizing *ad hoc* networks, [7] presented an algorithmic framework consisting of a large number of robots and small, cheap, simple wireless sensors to perform proficient and robust target tracking. Without dependence on magnetic compass or GPS localization service, they described a robotic sensor network for target tracking, focusing on algorithms which are simple for information propagation and distributed decision making. They presented a robotic sensor network system that autonomously conducts target tracking without component possessing localization capabilities. The approach provides a way out with minimal hardware assumptions (in terms of sensing, localization, broadcast, and memory/processing capabilities) while subject to a dynamic changing environment. Moreover the framework adjusts dynamically to both target movement and addition/deletion of network components. The network gradient algorithm provides an advantageous trade-off between power consumption and performance and requires relatively bandwidth [71]. The monitoring of EMF phenomena is extremely important in

practice, especially to guarantee the protection of the people living and working where these phenomena are significant. A specific robotic sensor network oriented to monitor electromagnetic fields (EMFs) was presented [8]. The activities of the system are being supervised by a coordinator computer, while a number of explorers (mobile robots equipped with EMF sensors) navigate in the environment and perform EMF measurement tasks. The system is a robotic sensor network that can autonomously deploy its explorers in an environment to cope with events like a moving EMF source. The system architecture is hierarchical. The activities of the system are being supervised by the computer and to perform the EMF tasks a number of explorers (mobile robots equipped with EMF sensors navigate through the environment). The grid map of the environment is maintained by the system, in which each cell can be either free or occupied by an obstacle or by a robot. The map is supposed to be known by the coordinator and the explorers. The environment is assumed to be static and the map is used by the explorers to navigate in the environment and by the coordinator to localize the EMF source [72].

5. Coverage for Multirobots

The use of multirobots holds numerous advantages over a single robot system. Their potential of doing work is way far better than that of single robot system [73]. Multiple robots can increase the robustness and the flexibility of the system by taking benefits of redundancy and inherent parallelism. They can also cover an area more quickly than a single robot and they have also potential to accomplish a single task faster than a single robot. Multiple robots can also localize themselves more efficiently when they have different sensor capabilities. Coverage for multirobot systems is an important field and is vital for many tasks like search and rescue, intrusion detection, sensor deployment, harvesting and mine clearing, and so forth [74]. To get the coverage the robots must be capable of spotting the obstacles in the environment and they should also exchange their knowledge of environment and have a mechanism to assign the coverage tasks among themselves [75]. The problem of deploying a mobile sensor network into an environment was addressed in [76] with the task of maximizing sensor coverage and also two behavior based techniques for solving the 2D coverage problems using multiple robots were proposed. Informative and molecular techniques are the techniques proposed for solving coverage problems and both of these techniques have the same architecture. When robots are within the sensor range of each other, the informative approach is to assign local identities to them. This approach allows robots to spread out in a coordinated manner because it is based on ephemeral identification where temporary local identities are assigned and mutual local information is exchanged. No local identification is made in molecular approach and also robots do not perform any directed communication. Each robot moves in a direction without communicating its neighbors because it selects its direction away from all its immediate sensed neighbors. Then these algorithms were compared with another approach known as basic approach, which only seeks

to maximize each individual robot's sensor coverage [74]. Both these approaches perform significantly better than basic approach and with the addition of few robots the coverage area quickly maximizes. An algorithm named (StiCo) which is an coverage algorithm was proposed for multirobot systems in [77]. This algorithm is based on the principle of stigmergic (pheromone-type) coordination known from the ant societies where a group of robots coordinate indirectly via ant-like stigmergic communication. This algorithm does not require any prior information about the environment and also no direct robot-robot communication is required. Similar kind of approach was used by Wagner et al. [78] for coverage in multirobot in which a robot deposits a pheromone which could then be detected by other robots; these pheromones come up with a decay rate, allowing continuous coverage of an area via implicit coordination [75]. For multirobot coverage, [75] proposed boustrophedon decomposition algorithm in which the robots are initially distributed through space and each robot is allocated at virtually bounded area to cover the area and is then decomposed into cells with the fixed cell width. By using the adjacency graph the decomposed area is represented which is incrementally constructed and shared among all robots and without any restriction robot communication is also available. By sharing information regularly and task selection protocol performance is improved. By planting laser beacons in environment the problem of localization in the hardware experiment was overthrown and using the laser range finder to localize the robots as this was the major problem to guarantee accurate and consistent coverage. Based on spanning-tree coverage of approximate cell decomposition, robustness and efficiency in a family of multirobot coverage algorithms was addressed [79]. Their approach is based on *offline* coverage: it is assumed that the robots have a map of the area a priori. The algorithm they proposed decomposes the work area into cells, where each cell is a square of size $4D$ and each cell is then further broken down into quadrants of size D .

6. Localization for Robots

In mobile robotics localization is a key component [80]. The process to determine the robots position within the environment is called localization or we can say that it is a process that takes a map as an input and estimates the current pose of the robot, a set of sensor readings, and then outputs the robot's current pose as a new estimate [81]. There are many technologies available for robot localization including GPS, active/passive beacons, odometer (dead reckoning), and sonar. For robot localization and map count an algorithm was presented using data from a range based sonar sensor [82]. For localization the robots position is determined by the algorithm by correlating a local map with a global map. Actually no prior knowledge of the environment is assumed; it uses sensor data to construct the global map dynamically. The algorithm estimates robots location by computing positions called feasible poses where the expected view of robot matches approximately the observed range sensor data. The algorithm then selects the best fit from the feasible poses. It requires robots orientation information to

make sure that the algorithm identifies the feasible poses. For location information Vassilis also used dead reckoning as a secondary source; when combined with range sensor based localization algorithm it can provide a close real time location estimate. A Monte Carlo localization algorithm was introduced using (MHL) was used for mobile robot position estimation [83]. They used the Monte Carlo type methods and then combined the advantages of their previous work in which grid based Markov localization with efficiency and accuracy of Kalman filter based techniques was used. MCL method is able to deal with ambiguities and thus can globally localize the robot. As compared to their previous grid based method MCL method has significantly reduced memory requirements while at the same time incorporating sensor measurements at a considerably higher frequency. Based on condensation algorithm the Monte Carlo localization method was proposed in [84]. It localizes the robot globally using a scalar brightness measurement when given a visual map of the ceiling. Sensor information of low feature is used by these probabilistic methods specifically in 2D plane and needs the robot to move around for probabilities to gradually converge toward a peak. The pose of the robots was also computed by some researchers based on the appearance. Panoramic image-based model for robot localization was used by Cobzas and Zhang [55]; with the depth and 3D planarity information the panoramic model was constructed, while the matching is based on planar patches. For probabilistic appearance based robot localization [56] used panoramic images. For extracting the 15-dimensional feature vectors for Markov localization PCA is applied to hundreds of training images. In urban environments the problem of mobile robot localization was addressed by Talluri and Aggarwal [85] by using feature correspondence between images taken by camera on robot and a CAD or similar model of its environment. For localization of car in urban environments, [86] used an inertial measurement unit and a sensor suite consists of four GPS antennas. Humanoid robots are getting popular as research tools as they offer new viewpoint compared to wheeled vehicle. A lot of work has been done so far on the localization for humanoid robots. In order to estimate the location of the robot [87] applied a vision based approach and then compared the current image to previously recorded reference images. In the local environment of the humanoid [88] detects objects with given colors and shapes and then determines its pose relative to these objects. With respect to a close object [89] localizes the robot to track the 6D pose of a manually initialized object relative to camera by applying a model based approach.

7. Conclusion

In this paper we reviewed MSN issues, sensor network applications in robotics and vice versa, robot localization, and also coverage for multiple robots. This is certainly not the extent that robotics is used in wireless sensor networks and also wireless sensor networks in robotics. However, we found that integrating static nodes with mobile robots enhances the capabilities of both types of devices and also enables new applications. The possibilities of robotics and wireless

sensor networks being used together seem endless and if used together in future also will help to solve many problems.

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgment

This work was supported by the Brain Korea 21 PLUS Project, National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) (no. 2013R1A2A2A01068127 and no. 2013R1A1A2A10009458).

References

- [1] S. Basagni, A. Carosi, and C. Petrioli, "Mobility in wireless sensor networks," *Journal of Wireless Networks*, vol. 14, no. 6, pp. 831–858.
- [2] C. Zhu, L. Shu, T. Hara, L. Wang, and S. Nishio, "Research issues on mobile sensor networks," in *Proceedings of the 5th International ICST Conference on Communications and Networking in China (ChinaCom '10)*, pp. 1–6, IEEE, Beijing, China, August 2010.
- [3] G. Song, Y. Zhou, F. Ding, and A. Song, "A mobile sensor network system for monitoring of unfriendly environments," *Sensors*, vol. 8, no. 11, pp. 7259–7274, 2008.
- [4] <http://www.wikipedia.com/>.
- [5] C. Flanagan, "A survey on robotics system and performance analysis," <http://www.cse.wustl.edu/~jain/cse567-11/ftp/robots/>.
- [6] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.
- [7] J. Reich and E. Sklar, "Robot-sensor Networks for search and rescue," in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, Gaithersburg, Md, USA, August 2006.
- [8] F. Amigoni, G. Fontana, and S. Mazzuca, "Robotic sensor networks: an application to monitoring electro-magnetic fields," in *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering*, pp. 384–393, 2007.
- [9] S. Shue and J. M. Conrad, "A survey of robotic applications in wireless sensor networks," in *Proceedings of the IEEE Southeastcon*, pp. 1–5, Jacksonville, Fla, USA, April 2013.
- [10] http://en.wikipedia.org/wiki/Sensor_node.
- [11] <http://www.merriam-webster.com/dictionary/sensor>.
- [12] <http://webs.cs.berkeley.edu/tos/>.
- [13] <http://www.pages.drexel.edu/~kws23/tutorials/motes/motes.html>.
- [14] E. Stavrou and A. Pitsillides, "Security evaluation methodology for intrusion recovery protocols in wireless sensor networks," in *Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 167–170, Paphos, Cyprus, 2012.
- [15] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01)*, vol. 3, pp. 1380–1387, April 2001.
- [16] B. Liu, O. Dousse, P. Nain, and D. Towsley, "Dynamic coverage of mobile sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 2, pp. 301–311, 2013.
- [17] J. Luo and Q. Zhang, "Probabilistic coverage map for mobile sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '08)*, pp. 1–5, New Orleans, La, USA, December 2008.
- [18] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, IEEE, 1985.
- [19] R. C. Arkin, "Motor schema-based mobile robot navigation," *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 92–112, 1989.
- [20] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem," in *Distributed Autonomous Robotic Systems 5*, chapter 8, pp. 299–308, Springer, Tokyo, Japan, 2002.
- [21] S. Poduri and G. S. Sukhatme, "Constrained coverage for mobile sensor networks," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '04)*, vol. 1, pp. 165–171, IEEE, May 2004.
- [22] A. Howard, M. J. Mataric, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, no. 2, pp. 113–126, 2002.
- [23] G. Wang, G. Cao, and T. F. La Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.
- [24] G. Wang, G. Cao, and T. la Porta, "Movement-assisted sensor deployment," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 2469–2479, March 2004.
- [25] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, pp. 1293–1303, April 2003.
- [26] M. A. Batalin, M. Rahimi, Y. Yu et al., "Call and response: experiments in sampling the environment," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 25–38, 2004.
- [27] B. Liu, P. Brass, and O. Doussie, "Mobility improves coverage of sensor networks," in *Proceedings of the 6th International Symposium on Mobile Adhoc Networking (MobiHoc '05)*, pp. 300–308, 2005.
- [28] A. Maxim and G. S. Batalin, "Coverage, exploration and deployment by a mobile robot and a communication network," in *Proceedings of International Workshop on Information Processing in Sensor Networks*, pp. 376–391, PaloAlto Research Center (PARC), Palo Alto, Calif, USA, April 2003.
- [29] G. Wang, G. Cao, T. La Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, vol. 4, pp. 2302–2312, Miami, Fla, USA, March 2005.
- [30] I. Amundson and X. D. Koutsoukos, "Mobile sensor localization and navigation using RF doppler shifts," in *Mobile Entity Localization and Tracking in GPS-less Environments: Second International Workshop, MELT 2009, Orlando, FL, USA, September 30, 2009. Proceedings*, vol. 5801 of *Lecture Notes in Computer Science*, pp. 235–254, Springer, Berlin, Germany, 2009.

- [31] Y. Ganggang and Y. Fengqi, "A localization algorithm for mobile wireless sensor networks," in *Proceedings of the IEEE International Conference on Integration Technology (ICIT '07)*, pp. 623–627, March 2007.
- [32] J. Yi, J. Koo, and H. Cha, "A localization technique for mobile sensor networks using archived anchor information," in *Proceedings of the 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '08)*, pp. 64–72, San Francisco, Calif, USA, June 2008.
- [33] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pp. 45–57, October 2004.
- [34] B. Dil, S. Dulman, and P. Havinga, "Range-based localization in mobile sensor networks," in *Wireless Sensor Networks: Third European Workshop, EWSN 2006, Zurich, Switzerland, February 13–15, 2006. Proceedings*, vol. 3868 of *Lecture Notes in Computer Science*, pp. 164–179, Springer, Berlin, Germany, 2006.
- [35] S. Tilak, V. Kolar, N. B. Abu-Ghazaleh, and K.-D. Kang, "Dynamic localization control for mobile sensor networks," in *Proceedings of the 24th IEEE International Performance, Computing, and Communications Conference (IPCCC '05)*, pp. 587–592, IEEE, April 2005.
- [36] S. Tilak, V. Kolar, N. B. Abu Ghazaleh, and K.-D. Kang, "Dynamic localization protocols for mobile sensor networks," <http://arxiv.org/abs/cs/0408042>.
- [37] B. Sau, S. Mukhopadhyaya, and K. Mukhopadhyaya, "Localization control to locate mobile sensors," in *Distributed Computing and Internet Technology: Proceedings of the 3rd International Conference (ICDCIT '06), Bhubaneswar, India, December 20–23, 2006*, vol. 4317 of *Lecture Notes in Computer Science*, pp. 81–88, Springer, Berlin, Germany, 2006.
- [38] C. Saad, A. R. Benslimane, and J.-C. Koing, "A distributed method to localization for mobile sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, 2007.
- [39] http://en.wikipedia.org/wiki/Positioning_technology.
- [40] <http://en.wikipedia.org/wiki/ZigBee>.
- [41] Z. Farid, R. Nordin, and M. Ismail, "Recent advances in wireless indoor localization techniques and system," *Journal of Computer Networks and Communications*, vol. 2013, Article ID 185138, 12 pages, 2013.
- [42] A. Popleteev, V. Osmani, and O. Mayora, "Investigation of indoor localization with ambient FM radio stations," in *Proceedings of PerCom-2012*, Lugano, Switzerland, March 2012.
- [43] http://compnetworking.about.com/od/networkprotocols/g/ultra_wide_band.htm.
- [44] C. Frost, C. S. Jensen, K. S. Luckow, B. Thomsen, and R. Hansen, "Bluetooth indoor positioning system using fingerprinting," in *Mobile Lightweight Wireless Systems*, vol. 81 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 136–150, Springer, Berlin, Germany, 2012.
- [45] http://en.wikipedia.org/wiki/Hybrid_positioning_system.
- [46] L. Niu, "A survey of wireless indoor positioning technology for fire emergency routing," *IOP Conference Series: Earth and Environmental Science*, vol. 18, Article ID 012127, 2014, Proceedings of the 8th International Symposium of the Digital Earth (ISDE '08).
- [47] A. Cheriet, M. Ouslim, and K. Aizi, "Localization in a wireless sensor network based on RSSI and a decision tree," *Przeglad Elektrotechniczny*, vol. 89, no. 12, pp. 121–125, 2013.
- [48] Y.-T. Chen, C.-L. Yang, Y.-K. Chang, and C.-P. Chu, "A RSSI-based algorithm for indoor localization using zigbee in wireless sensor network," in *Proceedings of the 15th International Conference on Distributed Multimedia Systems (DMS '09)*, pp. 70–75, 2009.
- [49] U. Ahmad, A. Gavrilov, U. Nasir, M. Iqbal, S. J. Cho, and S. Lee, "In-building localization using neural networks," in *Proceedings of the IEEE International Conference on Engineering of Intelligent Systems (ICEIS '06)*, April 2006.
- [50] L. Yu, "Fingerprinting localization based on neural networks and ultra wideband signals," in *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT '11)*, pp. 184–189, 2011.
- [51] C. Laoudias, D. G. Eliades, P. Kemppi, C. G. Panayiotou, and M. M. Polycarpou, "Indoor localization using neural networks with location fingerprints," in *Artificial Neural Networks—ICANN 2009*, vol. 5769 of *Lecture Notes in Computer Science*, pp. 954–963, Springer, Berlin, Germany, 2009.
- [52] C. Nerguizian, C. Despains, and S. Affès, "Indoor geolocation with received signal strength fingerprinting technique and neural networks," in *Telecommunications and Networking—ICT 2004: 11th International Conference on Telecommunications, Fortaleza, Brazil, August 1–6, 2004. Proceedings*, vol. 3124 of *Lecture Notes in Computer Science*, pp. 866–875, Springer, Berlin, Germany, 2004.
- [53] S. Kumar and S.-R. Lee, "Localization with RSSI values for wireless sensor networks: an artificial neural network approach," in *Proceedings of the International Electronic Conference on Sensors and Applications*, 2014, Paper d007.
- [54] S.-H. Fang and T.-N. Lin, "Indoor location system based on discriminant-adaptive neural network in IEEE 802.11 environments," *IEEE Transactions on Neural Networks*, vol. 19, no. 11, pp. 1973–1978, 2008.
- [55] D. Cobzas and H. Zhang, "Cylindrical panoramic image-based model for robot localization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '01)*, pp. 1924–1930, November 2001.
- [56] B. J. A. Kröse, N. Vlassis, and R. Bunschoten, "Omni directional vision for appearance-based robot localization," in *Sensor Based Intelligent Robots*, vol. 2238 of *Lecture Notes in Computer Science*, pp. 39–50, Springer, 2002.
- [57] N. Aida Mahiddin, E. Nadia Madi, S. Dhalila, E. Fadzli Hasan, S. Safie, and N. Safie, "User position detection in an indoor environment," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 8, no. 5, pp. 303–312, 2013.
- [58] J. Xu, W. Liu, F. Lang, Y. Zhang, and C. Wang, "Distance measurement model based on RSSI in WSN," *Wireless Sensor Network*, vol. 2, pp. 606–611, 2010.
- [59] G. Jekabsons, V. Kairish, and V. Zuravlyov, "An analysis of Wi-Fi based indoor positioning accuracy," *Scientific Journal of Riga Technical University, Computer Sciences*, vol. 44, no. 1, pp. 131–137, 2012.
- [60] S. Capkun, M. Hamdi, and J. P. Hubaux, "GPS free positioning in mobile ad hoc networks," *Journal Cluster Computing*, vol. 5, no. 2, pp. 157–167, 2002.
- [61] D. Djenouri, L. Khelladi, and N. Badache, "A survey of security issues in mobile ad hoc and sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 7, no. 4, pp. 2–28, 2005.

- [62] Y. Wang, G. Attebury, and B. Ramamurthy, "A survey of security issues in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 2, pp. 2–23, 2006.
- [63] V. Kumar, A. Jain, and P. N. Barwa, "Wireless sensor networks: security issues, challenges and solutions," *International Journal of Information and Computation Technology*, vol. 4, no. 8, pp. 859–868, 2014.
- [64] H. Kaur, "Attacks in wireless sensor networks," *Research Cell*. In press.
- [65] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Wormhole attacks in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 370–380, 2006.
- [66] G. Padmavathi and D. Shanmugapriya, "A survey of attacks, security mechanisms and challenges in WSN," *International Journal of Computer Science and Information Security*, vol. 4, no. 1-2, 2009.
- [67] A. LaMarca, W. Brunette, D. Koizumi et al., "PlantCare: an investigation in practical ubiquitous systems," in *UbiComp 2002: Ubiquitous Computing: 4th International Conference Göteborg, Sweden, September 29-October 1, 2002 Proceedings*, vol. 2498 of *Lecture Notes in Computer Science*, pp. 316–332, Springer, Berlin, Germany, 2002.
- [68] X. Li, I. Lille, R. Falcon, A. Nayak, and I. Stojmenovic, "Servicing wireless sensor networks by mobile robots," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 147–154, 2012.
- [69] S. M. Schaffert, *Closing the loop: control and robot navigation in wireless sensor networks [Ph.D. thesis]*, EECS Department, University of California, Berkeley, Calif, USA, 2006.
- [70] J.-P. Sheu, K.-Y. Hsieh, and P.-W. Cheng, "Design and implementation of mobile robot for nodes replacement in wireless sensor networks," *Journal of Information Science and Engineering*, vol. 24, no. 2, pp. 393–410, 2008.
- [71] J. Reich and E. Sklar, "Robotic sensor networks for search and rescue," in *Proceedings of the IEEE International Workshop on Safety, Security, and Rescue Robotics (SSRR '06)*, 2006.
- [72] F. Amigoni, G. Fontana, and S. Mazzuca, "Robotic sensor networks: an application to monitoring electro-magnetic fields," in *Proceedings of the Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real World AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, pp. 384–393, IOS Press, Amsterdam, The Netherlands, 2007.
- [73] L. Iocchi, D. Nardi, and M. Salerno, "Reactivity and deliberation: a survey on multi-robot systems," in *Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, vol. 2103 of *Lecture Notes in Computer Science*, pp. 9–32, Springer, Berlin, Germany, 2001.
- [74] B. Walenz, "Multi robot coverage and exploration: a survey of existing techniques," <http://bwalenz.files.wordpress.com/2010/06/csci8486-walenz-paper.pdf>.
- [75] S. K. Chan, A. P. New, and I. Rekleitis, "Distributed coverage with multi-robot system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '06)*, pp. 2423–2429, Orlando, Fla, USA, May 2006.
- [76] M. A. Batalin and G. S. Sukhatme, "Spreading out: a local approach to multi-robot coverage," in *Proceedings of the 6th International Symposium on Distributed Autonomous Robotics System*, pp. 373–382, Fukuoka, Japan, June 2002.
- [77] B. Ranjbar-Sahraei, G. Weiss, and A. Nakisae, "stigmergic coverage algorithm for multi-robot systems (demonstration)," in *Proceedings of the 10th German conference (MATES '12)*, pp. 126–138, Springer, Trier, Germany, October 2012.
- [78] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein, "Distributed covering by ant-robots using evaporating traces," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 5, pp. 918–933, 1999.
- [79] N. Hazon and G. A. Kaminka, "On redundancy, efficiency, and robustness in coverage for multiple robots," *Robotics and Autonomous Systems*, vol. 56, no. 12, pp. 1102–1114, 2008.
- [80] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.
- [81] R. G. Brown and B. R. Donald, "Mobile robot self-localization without explicit landmarks," *Algorithmica*, vol. 26, no. 3-4, pp. 515–559, 2000.
- [82] V. Varveropoulos, Robot Localization and Map construction using sonar data, The Rossum project.
- [83] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Robust Monte Carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.
- [84] F. Dellaert, W. Burgard, D. Fox, and S. Thrun, "Using the condensation algorithm for robust, vision-based mobile robot localization," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '99)*, pp. 588–594, June 1999.
- [85] R. Talluri and J. K. Aggarwal, "Mobile robot self-location using model-image feature correspondence," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 63–77, 1996.
- [86] R. Nayak, "Reliable and continuous urban navigation using multiple Gps antenna and a low cost IMU," in *Proceedings of the ION GPS*, Salt Lake City, Utah, USA, September 2000.
- [87] J. Ido, Y. Shimizu, Y. Matsumoto, and T. Ogasawara, "Indoor navigation for a humanoid robot using a view sequence," *International Journal of Robotics Research*, vol. 28, no. 2, pp. 315–325, 2009.
- [88] R. Cupec, G. Schmidt, and O. Lorch, "Experiments in vision-guided robot walking in a structured scenario," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '05)*, pp. 1581–1586, Dubrovnik, Croatia, June 2005.
- [89] P. Michel, J. Chestnutt, S. Kagami, K. Nishiwaki, J. Kuffner, and T. Kanade, "GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '07)*, pp. 463–469, IEEE, San Diego, Calif, USA, November 2007.

Research Article

A Self-Adaptive Wireless Sensor Network Coverage Method for Intrusion Tolerance Based on Trust Value

Zuo Chen,^{1,2} Xue Li,¹ Bing Yang,³ and Qian Zhang¹

¹College of Computer Science and Electronic Engineering, Hunan University, Changsha, Hunan 410082, China

²Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

³School of Education, Hubei University, Wuhan, Hubei 430062, China

Correspondence should be addressed to Bing Yang; yangbing@126.com

Received 18 August 2014; Accepted 8 October 2014

Academic Editor: Fei Yu

Copyright © 2015 Zuo Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The sensor is quite easily attacked or invaded during the process of the node coverage optimization. It is a great challenge to make sure that the wireless sensor network could still carry out a secure communication and reliable coverage under the condition of being attacked. Therefore, this paper proposes a network coverage method for intrusion tolerance based on trust value of nodes by combining the trust value model with the reliable coverage optimization. It first estimates trust value of nodes through which to regulate the perception radius and decision-making radius. Furthermore, this algorithm also combines the classical methods of wireless network coverage, such as GSO and PSO, to realize the networks coverage of invasive tolerant sensor. After comparing with the conventional single cover mechanism, it can improve the security and coverage rate of network under the condition of invasion. The simulation results verify the effectiveness of the algorithm.

1. Introduction

With the progress of science and technology in recent years, wireless sensor has a trend towards miniaturization, high efficiency, low power consumption, and so on and also can be applied to the field of radio communication to achieve cost-effectiveness in mass production. Wireless sensor network node generally spreads to the region where people could not reach; it can organize itself as a network, search router, and detect the surrounding environment [1]. Due to the characteristics restriction of sensor node and the uncertain natural environment, the reliability and security of data is particularly important. Wireless sensor network, which can provide reliable communication and coverage under the condition of resource constraints and inevitable vulnerability along with the attacks and damage behavior, is urgently required.

In order to save energy, the researchers began to integrate coverage control function into the traditional sensor, so as to use fewer nodes as much as possible to complete the requirement of coverage. Regarding the problem of WSN node with vulnerable attacks and insecurity, the concept of trust was

presented by the researchers to ensure the safety of the coverage area. At present, security routing technology based on trust management [2], safe fusion technology [3], and security time synchronization technology [4] have achieved some valuable research results, but up to now, the research field has little effective security cover mechanism based on trust management.

To improve the quality of coverage with the traditional covering algorithm and reduce effects of fault node on quality of coverage at the same time, researchers have worked out some reliable coverage algorithms. The literature [5] monitors the behavior of discard packets through some extra monitoring nodes in a network and helps failure nodes to send packets and support the sensor to transfer data to the sink node safely and accurately. The literature [6] proposed a reliable cover mechanism based on 2-Coverage, by increasing the redundancy cover to block out the single node failure, and therefore it can achieve the fault-tolerant effect. In order to avoid a single node failure which may cause the movement of whole mobile network nodes, literature [7] presented a scheduling scheme based on virtual coordinate. These studies

of reliable cover mechanism mentioned above are mainly focused on the solution to reduce failure nodes on the quality of the cover. However, in many real situations, in addition to the node failure, network attack and invasion will also cause the decrease of the quality of the network coverage. For example, if the monitoring nodes in the network were captured by the enemy, turned into the malicious nodes, then the strengthening auxiliary mechanism of monitoring node will lose effectiveness. If part of the network was invaded, sensor data will gravely deviate from the real data [8], and these false data will probably lead to false alarms and consumption of limited network resources and cause serious consequence. So the researchers put forward the trust management mechanism, by monitoring the behavior of nodes to calculate the trust value of node, determine the behavior of the next term according to the trust. Trust management model is one of the effective methods to guarantee the network security in the case of wireless sensor network breach.

This paper proposes a coverage method for the invasive tolerant and adaptive sensor network based on trust value. Based on the coverage optimization in combination with the trust model, trust evaluation mechanism is able to identify malicious nodes and no cooperative nodes, resist network attack and invasion, improving coverage, and ensure the safety coverage of WSN.

The rest of this paper is organized as follows; the second part presents the system model; in the third part, trust management model is introduced and described; the fourth section presents the details of algorithm process; the fifth part is for simulation and analysis of simulation results; the last part is summary and outlook.

2. System Model

The coverage method for the invasive tolerant and adaptive sensor network based on trust value in this paper mainly includes two phases: the evaluation of node trust and the network coverage optimization. In the trust evaluation stage, the trust value of each node is calculated according to their own information and neighbor node information integration. In network coverage optimization stage, the node trust value is mapped to the radius changes and the traditional optimization of network coverage algorithm is applied to maximize reliable coverage.

2.1. Network Model. Suppose N sensor nodes were randomly distributed in the $M * M$ square area and sensor node has key features as follows.

- (1) All of the initial position and velocity of sensor nodes vary within a certain range of random initialization.
- (2) Only one BS in the entire network node and BS node's energy is infinite.
- (3) All sensor nodes in the network communication model and cognitive model are disc model; the original decision radius is twice the radius of perception, and all the radiuses are adjustable.

- (4) Sensor nodes can be moved after deployment.
- (5) Node is not equipped with GPS, but each node can know the current position information of itself.
- (6) Each node establishes a trust table and a radius table for its neighbor nodes and records the change of trust value and the radius.

2.2. Network Coverage Model. C' is a subset of C , and $A(C')$ expresses the cover area of node set C' , and regional coverage could be defined as $P_{cov} = A(C')/A(C)$. In practice, directly calculating the value of $A(C)$ is complex; to simplify the calculation, detection area A will grid into $M * N$ points, the coordinates of the lattice are (x, y) , the distance from grid point to its sensor is defined as $d(c_i, p) = \sqrt{(x - x_i)^2 + (y - y_i)^2}$. Define the probability of grid points covered by sensor nodes as

$$p(x, y, c_i) = \begin{cases} 1 & \text{if } d(c_i, p) \leq r \\ 0 & \text{else.} \end{cases} \quad (1)$$

When grid point (x_k, y_k) covered by more than one sensor, mark it as covered state, and the probability of grid points been covered is defined as $z_k = \cup_{c'} p(x_k, y_k, c_i)$, so the value of z_k could only be 1 or 0. $A(C') = \sum_{M \times N} z_k$ and $A(C) = M \times N$, so area coverage is defined as

$$P_{cov} = \frac{A(C')}{M \times N}. \quad (2)$$

3. Trust Management Model

Comparing with the traditional security mechanism, trust management model has much more flexibility, scalability, and reliability. Meanwhile, it can complete the reliable authorization management, resource sharing, and security service via its establishment of a trust relationship among entities. Trust model is the main assessment according to "trust," by assessing behavior observation and interaction of individual record information, to get the evaluation of individual trust value computing model using appropriate mathematical calculation model. The trust model can be used to improve the security in open network environment based on the calculation and evaluation of node trust values for the implementation of the flexible adjustment of network security policy.

To this day, there are a lot of trust models for wireless sensor network (WSN) research. There are some secure routing technologies based on trust model. Wang et al. [9] point out a kind of wireless sensor network security routing algorithm without fixed infrastructure and with hardly detected malicious behavior which covers the safe trouble. The algorithm introduces the node credibility and also can establish secure routing and eliminates the malicious nodes of the network. Zhang et al. [10] put forward a credible wireless sensor network routing algorithm based on subjective logic. The algorithm takes full consideration of the node's credibility when established in route choice,

to ensure the security of data transmission path constraint. Cheng et al. [11] put forward BT SR: an algorithm based on credible safety data fusion and routing. Respectively, from the perspective of the time and spatial correlation, it established a credible model based on similarity. This method solves the selective forwarding attacks and flood attacks in the process of information transmission to ensure the security of data transmission.

At present, there are few research results about reliable wireless sensor network covering algorithm. Shuhao and Xiaolon [12] and other researchers proposed an adaptive scheduling algorithm based on rotation of the trust nodes of the mesh. The algorithm is based on trust nodes scheduling which adapts rotation scheduling by the credibility of the nodes. Then find the virtual grid points that mesh formed, and quantify the trust value of each grid point. If the trust value cannot meet the requirements of safety cover, rescheduling to related nodes. After the calculation of trust value of the node for this algorithm, it should in turn calculate quantified credit of the trust grid points and thus become highly complicated. As the wireless sensor network typically consists of many small sensor nodes, and these sensor nodes only have limited communication bandwidth and energy, minimizing the complexity of the security covering algorithm can reduce energy consumption.

The trust model proposed in this paper takes different trust factors into consideration, including the observation from subjective one to the objective one and the recommendation of the third party. Neighbor nodes monitor each other, according to direct and indirect trust value from subjective one to the objective one, and then get comprehensive trust value. By adjusting the size of perception radius and decision radius according to comprehensive trust value to reduce the decision-making area and the coverage area of the low credibility nodes, in order to ensure the reliable coverage of the network.

3.1. Definition of Trust Factors. Assuming nodes i and j are neighbor nodes, node i evaluates the trust of node j from the aspects such as information communication, data integrity and consistency, and quantitative analysis of the various factors influencing the trust value.

3.1.1. The Consistency Factor $CF_{i,j}(t)$. To prevent malicious nodes forged packets, it is needed to analyze the space data consistency of the adjacent node. In wireless sensor networks, local neighbor nodes' data generally have high correlation. Subject i monitored the packet content of object j , and comparing with its acquisition of data, if the difference of the two within a certain range, we can think that the assessment for monitoring objects has consistency between the subject and object. The consistency factor is as follows:

$$CF_{i,j}(t) = \frac{CP_{i,j}(t)}{CP_{i,j}(t) + NCP_{i,j}(t)}. \quad (3)$$

$CP_{i,j}(t)$ is the number of consistent data packets, and $NCP_{i,j}(t)$ is the number of inconsistent data packets.

3.1.2. Sending Rate Factor $SF_{i,j}(t)$. Subject i evaluates and monitors the data sending situation of object j ; if packet number is lower than the threshold limit TL , we can think it as a selfish node, and if the packet amount exceeds the maximum limit threshold TH , we can think that it was in denial of service attack. Sending rate factor is as follows:

$$SF_{i,j}(t) = \begin{cases} \frac{SP_{i,j}(t) - T_L}{ES_{i,j}(t) - T_L}, & SP_{i,j}(t) < ES_{i,j}(t) \\ \frac{T_H - SP_{i,j}(t)}{T_H - ES_{i,j}(t)}, & SP_{i,j}(t) \geq ES_{i,j}(t). \end{cases} \quad (4)$$

$SP_{i,j}(t)$ is the number of transmitting data packets within the period t , $ES_{i,j}(t)$ is the expected value of the total transmitted data packets within the period t , and it was preset by the base station according to the application. When sending rate factor is in the ideal range, node trust has a higher value.

3.1.3. Integrity Factor $IF_{i,j}(t)$. To prevent malicious nodes tampering and forwarding packets, we need to evaluate the integrity of the packets. After the source node sends data packets in a certain time, we monitor whether the next-hop node performed data forwarding correctly. Integrity factor is as follows:

$$IF_{i,j}(t) = \frac{IP_{i,j}(t)}{DTP_{i,j}(t)}. \quad (5)$$

$IP_{i,j}(t)$ is the complete forward package number and $DTP_{i,j}(t)$ is the number of packets that subject i needs object j to forward.

3.1.4. Radius Factor $RA(t)$. The trust between nodes boils down to whether or not to transmit data packets. In order to prevent malicious nodes transmit more non-normal data, right amount to reduce the size of perception radius and decision radius of low trust value nodes. Radius factor is as follows:

$$RA(t) = \frac{R(t)}{R_0}. \quad (6)$$

$R(t)$ is the radius of node in the period t and R_0 is original radius.

3.1.5. The Time Factor $TF(t)$. Due to that fact that node trust value is combination of trust record and current observation information, the time factor could be joined to analyze the correlation with the context of the trust value, to reflect trust value with time attenuation effect. If the time factor is too large, the trust value is affected by the history too much and the evaluation of the node might be wrong, while if the time factor is too small, the trust value may have excessive dependence on a single time period. Therefore, we need to make different time factor according to different security levels.

3.2. The Calculation of Trust Value. According to the specific application requirements, evaluating node i monitoring part of or all of the trust factors, evaluate object j 's direct trust

values of $DT_{i,j}(T)$ by method of weighted average. In all of the defined trust factors, $SF_{i,j}(T)$ and $CF_{i,j}(T)$ mainly consider the rationality of the $IF_{i,j}(t)$ amount of data packet transmission and related content. They were involved in evaluation object data forwarding, part of forwarding the data packet integrity, and relative surplus energy. Suppose that evaluation of the trust value monitoring all trust factors mentioned above, and the historical trust of the previous cycle expressed as $DT_{i,j}(T-1)$ and the corresponding radius is $RA(T-1)$, calculation formula of direct trust value can be defined as

$$\begin{aligned} DT_{i,j}(t) &= TF(t) \\ &\times (\omega_1 SF_{i,j}(t) CF_{i,j}(t) + \omega_2 IF_{i,j}(t) + \omega_3 RA(t-1)) \\ &+ (1 - TF(t)) DT_{i,j}(t-1). \end{aligned} \quad (7)$$

$\omega_1, \omega_2, \omega_3$ are the weighted coefficients, which could be adjusted according to specific circumstances, and

$$\omega_1 + \omega_2 + \omega_3 = 1. \quad (8)$$

The interaction between the subject of i and the object of j is not only direct but also indirect through the common neighbor. So the node i trust calculation includes direct trust and indirect trust value. Indirect trust is limited to i and j common neighbor node trust transfer. The indirect trust value is as follows:

$$IT_{i,j}(t) = \frac{\sum_{k=1}^s DT_{i,k}(t) DT_{k,j}(t)}{s}. \quad (9)$$

k is one of i and j 's common neighbor nodes and s is the total number of the common neighbor nodes.

So the evaluation of i comprehensive trust degree on object j $AT_{i,j}(t)$ is as follows:

$$AT_{i,j}(t) = \lambda DT_{i,j}(t) + (1 - \lambda) IT_{i,j}(t). \quad (10)$$

λ is a comprehensive trust in direct trust degree of dependence and it can be adjusted for specific application.

3.3. Node Radius Adjustment of the Trust. With the assessment of trust between nodes, we can accurately judge the malicious nodes in the network. In wireless sensor networks, in order to mitigate what the malicious node brings, the sensing radius and the radius of decision nodes should be regulated according to the trust degree of node, in order to make the malicious nodes communication and the coverage reduction in the network. Each node can establish a radius table to its neighbor node, to record their radius change. When a node j was judged to be malicious nodes, the neighbor node will update its radius as follows:

$$R' = R_0 f \left(\frac{\sum_{i=1}^n AT_{i,j}(t)}{n} \right). \quad (11)$$

R_0 is the original radius, including sensing radius and radius of coverage. Node i is the neighbor of node j ; n is the total number of neighbors. Function f can be changed according to the need of different network while we used the linear function in this paper. After updating the radius, the radius of neighbor nodes list will also be updated.

4. Method Description

Swarm intelligence algorithm [13] is inspired by the nature of biological behavior, such as particle swarm optimization, ant colony optimization, and firefly algorithm. They have their own unique strengths but also have their own defects. In order to overcome the defect of the original algorithm, combining two or more algorithms together has become the trend of the research.

4.1. The Basic PSO and GSO Algorithm. In 1995, American psychologist Kennedy and electrical engineer Eberhart introduced an algorithm of particle swarm optimization (PSO) [14]. It is inspired by the migration and clustering of the birds during their foraging. In the original algorithm, its argument is consistent, and there is limitation in the solving process. Considering the basic particle swarm optimization, Jianping et al. [15] introduced inertia weight ω , which decreases linearly in the original formula, and formulated basic PSO algorithm. The algorithm [16] supports that every individual is a particle without volume and quality, flying at a certain speed in the search space, and adjusts its velocity dynamically according to the comprehensive analysis of the flying experience of individual and group.

In 2005, Luo et al. promoted a new swarm intelligence heuristic computing technology: GSO (glowworm swarm optimization) [17]. GSO algorithm developed a multipoint parallel global random search strategy based on the behavior of group [18]. High speed and efficiency of capturing the extreme point make GSO have strong versatility [19]. The significant factor of finding the optimization in firefly algorithm is the brightness and attraction. Brightness depends on the current position and the objective function value, which is higher when the position is better. At the same time, the attraction is influenced by the brightness, which means the brighter fireflies have the stronger attraction and also can attract those less bright fireflies. With the increase of distance, the brightness and attraction of media decrease after absorbing fluorescence. In all, there are four stages in the GSO process [20]: fluorescein update, firefly movement, firefly position update, and firefly neighbor radius update.

4.2. Improved Algorithm PGSO. PGSO adopts the method of series which combines particle swarm optimization (PSO) algorithm and the firefly algorithm. Firefly algorithm is able to discover global optimal solutions and local optimal solutions of the search space; the disadvantage is the high time cost and accuracy is not high. Particle swarm optimization has the advantage of fast convergence speed and high calculation accuracy; its shortcoming is ease of falling into local optimum. After merger of two algorithms, precision

of solution is higher than particle swarm optimization and firefly algorithm and helps to overcome the problem of falling into local optimum.

Assuming the improved algorithm PGSO's particle swarm is made up of m particle. Particle target search space is composed of n fireflies. $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ is the position vector of the i th particle ($i = 1, 2, \dots, m$); $v_i = (v_{i1}, v_{i2}, \dots, v_{in})$ is the i th particle velocity, on behalf of the next iteration particle moving distance; $p_i = (p_{i1}, p_{i2}, \dots, p_{in})$ is the optimal value in the search history of the i th particle, namely, the local optimal value; $p_g = (p_{g1}, p_{g2}, \dots, p_{gn})$ is the optimal value in the search history of the whole particle swarm, namely, the global optimal value. Each iteration process is as follows.

- (1) In the iterative optimization process, firstly, update the particle velocity and position based on particle swarm optimization (PSO) algorithm as follows:

$$v_{id}^{(k+1)} = \omega v_{id}^k + c_1 r_1 (p_{id} - x_{id}^k) + c_2 r_2 (p_{gd} - x_{id}^k) \quad (12)$$

$$x_{id}^{(k+1)} = x_{id}^k + v_{id}^{(k+1)}. \quad (13)$$

$i = 1, 2, \dots, m$; $d = 1, 2, \dots, n$; d is the current dimension for target search space; k is the current number of iterations; r_1 and r_2 are random numbers in $[0, 1]$ which obey uniform distribution; c_1 and c_2 are artificial learning factors. Inertia weight ω is as follows:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{\text{iter}_{\max}} \times k. \quad (14)$$

ω_{\max} is the initial weights; ω_{\min} is the weight of termination; iter_{\max} is maximum number of iterations, and k is the current number of iterations.

- (2) Applying the firefly algorithm iterative to update searching space. The firefly luciferin is updated as follows:

$$l_i(t+1) = (1 - \rho) l_i(t) + \gamma J(x_i(t+1)). \quad (15)$$

$l_i(t)$ is the fluorescein concentration of firefly i with t iteration; ρ ($0 < \rho < 1$) is the fluorescein concentration attenuation coefficient; $J(x_i(t))$ is the objective function values of node i with t iteration. The objective function is based on the coordinates of node i . Take the objective function as

$$J(x_i(t+1)) = \sum_{j=1}^k \frac{l_j(t+1)}{d_{ij}(t+1)}. \quad (16)$$

In the formula

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (17)$$

$$j \in N_i(t+1) = \{j : d_{ij}(t+1) < r_d^i(t+1), \|x_j(t) - x_i(t)\| > 0\}.$$

d_{ij} is the Euclidean distance between nodes i and j and j is the adjacent nodes of node i which is in the perception of radius and different to i . $r_d^i(t)$ is the radius of the adjacent decision domain of node i with t iteration. According to the formula (13) we update the neighbors fireflies' storage of $r_d^i(t)$, narrowing the scope of low trust fireflies decision making.

- (3) Computing the probability of firefly move to their neighbor whose fluorescein concentration is lower than itself we have

$$P_{ij} = \frac{l_i(t) - l_k(t)}{\sum_{k \in N_i(t)} (l_i(t) - l_k(t))}. \quad (18)$$

In the formula

$$k \in N_i(t) = \{k : d_{ik}(t) < r_d^i(t), l_i(t) < l_k(t)\}. \quad (19)$$

K is the adjacent nodes of node i whose fluorescein concentration is lower than i and in the perceived radius of i with t iteration. $d_{ik}(t)$ is the Euclidean distance between i and k with t iteration.

- (4) Location updates after the firefly moved is

$$x_i(t+1) = x_i(t) + s * \left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right). \quad (20)$$

$x_i(t)$ is the space position of node i in n dimension and s is step length of location update iteration. $\|x_j(t) - x_i(t)\|$ is Euclidean distance.

- (5) Updating each particle, nest the firefly algorithm in particle swarm optimization algorithm to keep accurate particle. After updating m particles, update individual optimal values and global optimal value according to the coverage.

4.3. The Intrusion Tolerance Security Coverage Method Based on PGSO Algorithm. When the network is under attack, or part of the node is invaded, sensor data gravely deviates from the real data and even breaks the authenticity of the data and may also infect neighbor nodes, consume limited network resources, and cause a serious consequence. The intrusion tolerance security coverage method is based on PGSO algorithm by using the above mentioned trust management model and combining PGSO iterative process to achieve the effect of adaptive adjustment cover as follows.

- (a) After initialization of m particles, according to formula (10) in trust models, calculate each node's trust, again by formula (11) to adjust the radius list of the neighbor node. Reduce the low trust nodes' coverage areas.
- (b) According to formula (11) to update the $r_d^i(t)$ stored in the neighbor nodes' list, narrow the scope of low trust fireflies decision making.
- (c) According to formula (10) compute node's trust after the update particle round. Regulate the nodes' radius and coverage according to formula (11) and then start a new iteration 3.

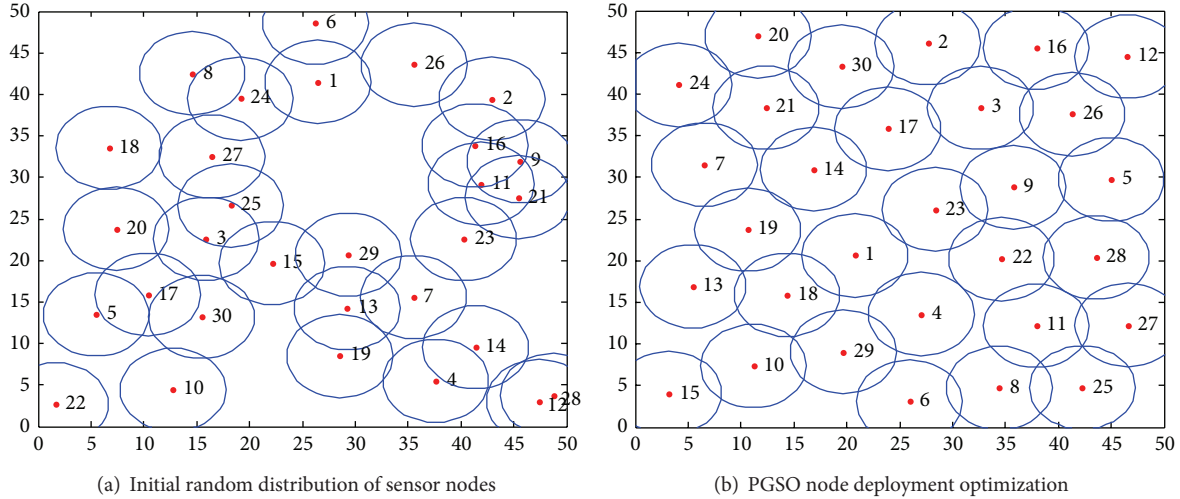


FIGURE 1: PGSO to 30 nodes deployment optimization algorithm.

5. Simulation

5.1. PGSO Algorithm Simulation and Performance Analysis.

In order to verify the validity of the PGSO algorithm, we evaluate the algorithm in MATLAB environment, so that we may improve the algorithm itself through the related simulation test and comparison. Set the particle size of particle swarm optimization (PSO) algorithm as $m = 40$, the number of nodes in each particle as $n = 30$; that is, the number of the fireflies is 30, the largest number of iterations is $iter_{max} = 200$, the initial value of perceived radius is 5 m, the initial value of radius of decision is 10 m, and learning factor is $c_1 = c_2 = 2$. Linear decreasing inertia weight ω , the initial weights $\omega_{max} = 0.9$, weight of the termination $\omega_{min} = 0.4$, the attenuation coefficient of fluorescein in GSO algorithm $\rho = 0.9$, fitness extraction ratio $\gamma = 0.1$. In the $50\text{ m} \times 50\text{ m}$ square monitoring area, mesh point size is set as $0.5\text{ m} \times 0.5\text{ m}$. Distribute 30 mobile sensor nodes in the monitoring area randomly and use PGSO algorithm to do the simulation. (a) and (b) in Figure 1 present node distribution simulation diagram before and after optimization, respectively.

As can be seen from the above, the network coverage strategy put forward in this paper can give more reasonable node deployment optimization scheme, improving the network coverage at the same time. From the simulation results, we can see node deployment from the complicated state optimal operation to the uniform distribution, and overlapped covered area by each other is relatively small. In order to further verify the validity of the algorithm, under the experimental environment, we, respectively, do the experimental simulation for PSO algorithm, GSO algorithm, and PGSO optimization algorithm. The simulation results are as shown in Figure 2.

As can be seen from Figure 2, network coverage of PGSO optimization algorithm in each iteration step is always greater than both the GSO algorithm and PSO algorithm. Furthermore, in order to test whether PGSO can give better network coverage optimization, under the different condition

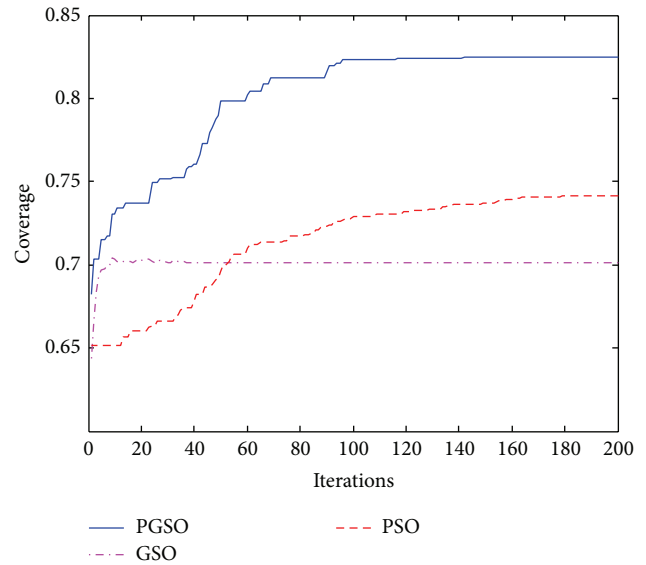


FIGURE 2: Network coverage comparison chart.

of the network node number, we set node numbers 10, 15, 20, 25, 30, 35, 40, and 45, separately, and adopt the GSO algorithm, PSO algorithm, and PGSO algorithm to make simulations. These results of each algorithm under different node number were taken using average network coverage of 10 times of simulation. The simulation results are shown in Figure 3. It can be seen from Figure 3, when the network node is at the same time, coverage of PGSO algorithm is always greater than that of GSO and PSO algorithms, and advantage will be more apparent especially in the larger network.

5.2. PGSO Algorithm Based on Trust Instance Simulation and Performance Analysis. This paper joins the trust model and PGSO to realize reliable coverage. In order to verify its

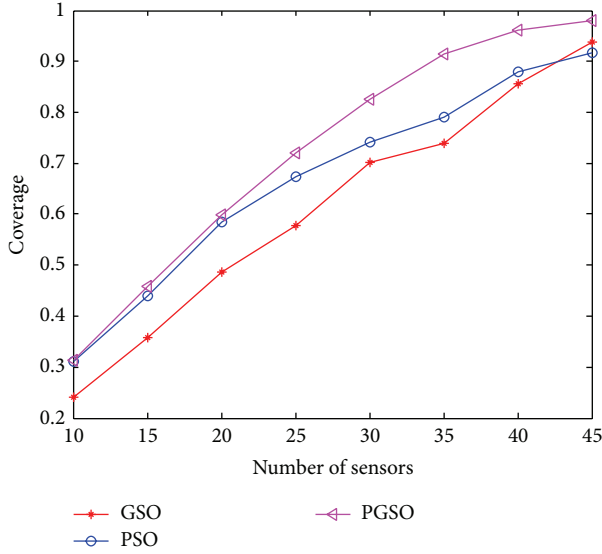


FIGURE 3: Coverage changes along with the network size.

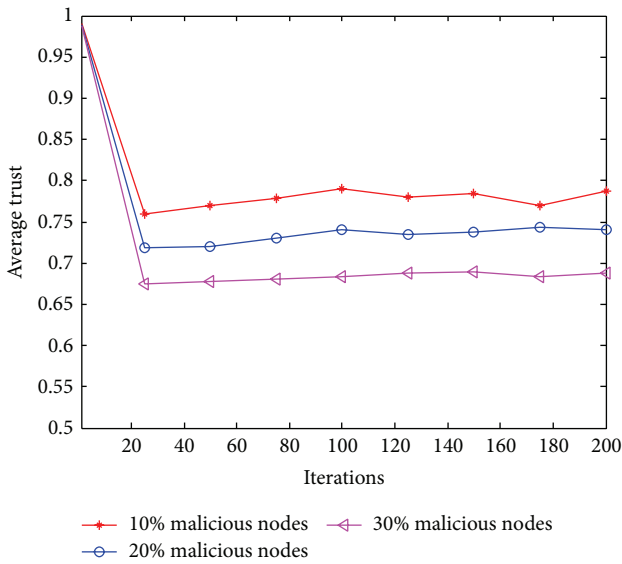


FIGURE 4: Average trust value changes of network node.

effectiveness, we take evaluated node trust value into account including trust factor such as data consistency, sending rate, integrity, and time. Set the particle size as $m = 40$; each particle has a number of nodes $n = 30$. A different number of nodes turn into malicious nodes by artificial random setting, and malicious nodes will send false data information or tamper with the forwarding packets. Monitor the change of trust value and the radius of normal nodes, malicious nodes, and network coverage in the meantime. By setting the malicious nodes proportion as 10%, 20%, and 30%, respectively, in the simulation experiment, node average trust values are shown in Figure 4. All nodes trust values are set to 1 at the beginning of the figure; that is, all the nodes are considered credible. By detecting average trust value of nodes

TABLE 1: Comparison of coverage in different network under varying invasion degree.

Algorithm	Invasion ratio		
	10%	20%	30%
PGSO1 (30)	0.7595	0.7101	0.6411
PGSO2 (30)	0.7718	0.7232	0.6559
PGSO1 (40)	0.8611	0.8283	0.7617
PGSO2 (40)	0.8977	0.8635	0.8112
PGSO1 (50)	0.9299	0.8855	0.8314
PGSO2 (50)	0.9474	0.9037	0.8525

in the network every 25 times of the iterative process, the average trust value of nodes tends to be stable during the subsequent iteration process. The average node trust value is lower when there are more intrusion nodes in the network.

Under the condition of the experimental environment mentioned above, a simulation of a set of comparative experiments is necessary. Contrast test does not include the node trust mechanism model, because in many cases, in order to guarantee the security of the network, once it detects malicious nodes, the malicious nodes will be placed as a dormant state or isolated state. For the sake of clearness, PGSO1 is short for the malicious node dormancy or segregated algorithm and PGSO2 is short for PGSO algorithm based on trust model. Set the proportion of malicious node as 10%, 20%, and 30%, respectively, in the simulation experiment. The simulation results are as shown in Figures 5 and 6.

In Figure 5, black triangle mark is invaded node, and the red dot is normal node. From a series of contrast figures, it can be found obviously that the coverage method for the invasive tolerant and adaptive sensor network based PGSO is much better. It has a larger coverage and also can ensure the security of network.

Seen from Figure 6, network coverage of PGSO2 algorithm in each iteration step is greater than that of PGSO1 basically. And in the case of node invasion increase, the drop of coverage is much smaller than that of isolated model. To further show that the coverage method for the invasive tolerant and adaptive sensor network based on trust value can give a more reasonable network coverage optimization scheme, consider the condition of the network node number is 30, 40, and 50. Respectively use PGSO1 and PGSO2 to simulate 10 times, and take the average network coverage into comparison. The simulation results are shown in Table 1. PGSO1 (30) in Table 1 expresses the PGSO1 algorithm simulation result when the node number of network is 30. And by this analogy, it can be seen from the data in the table that with the increase of network scale, PGSO2 has better performance than PGSO1. Therefore, PGSO2 has a tendency to do better in a large network.

6. Conclusion and Future Works

In order to guarantee the security of network under the maximum effective coverage, this paper proposes a network coverage method for the invasive tolerant sensor based

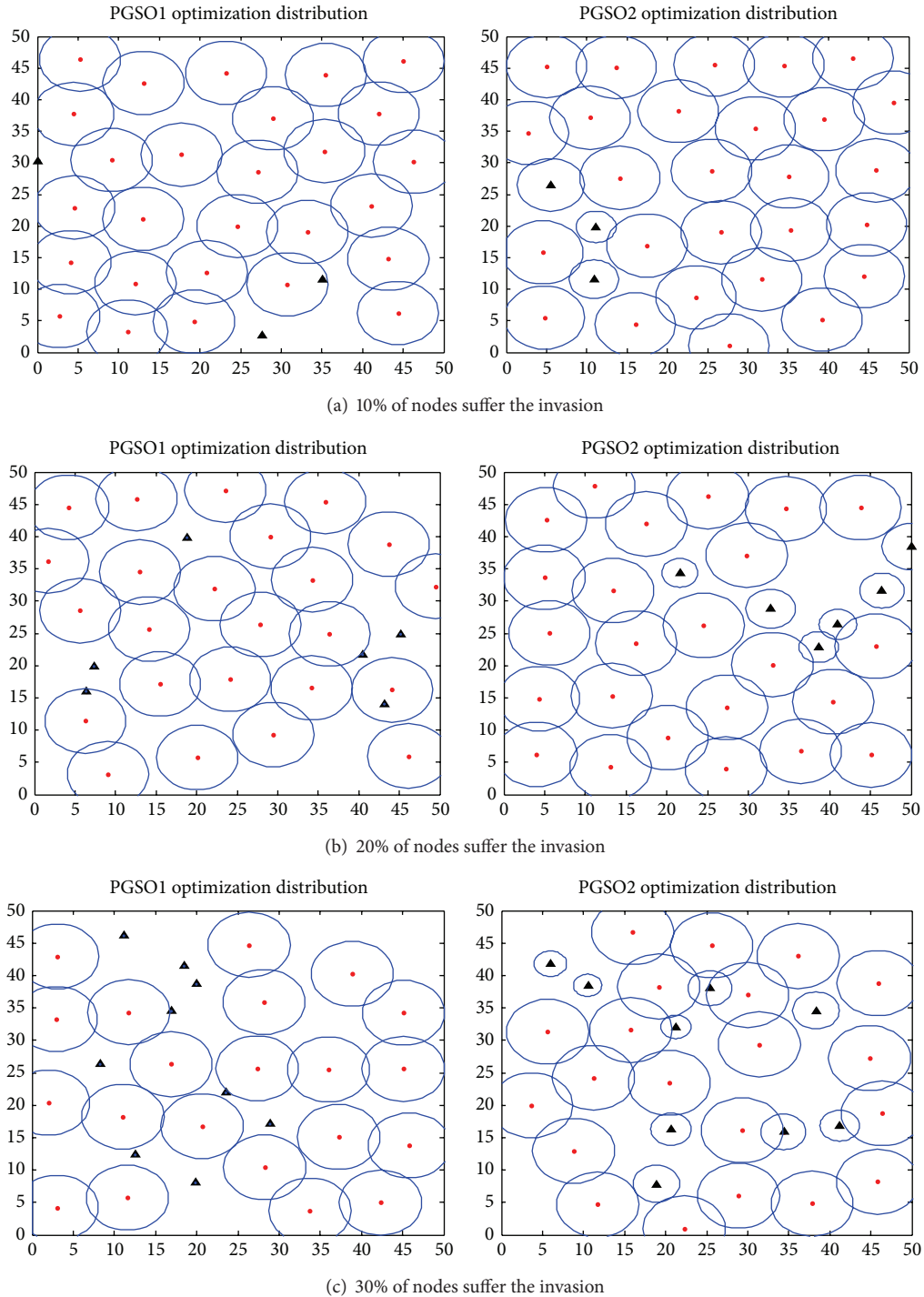


FIGURE 5: Contrast of algorithm optimization.

on trust value of nodes. Based on the consistency, data transmission rate, integrity, and time factor, nodes could estimate each other's trust value and next adjust radius of node according to the different trust value and, at the same time, integrate this trust model into the optimization algorithm of traditional coverage method, PGSO, making sure the network can adjust to the maximum reliable coverage

when under invasion. The simulation results show that the coverage rates of PGSO algorithm and GSO algorithm with PSO both have increased evidently. And based on the same reliable coverage rate, compared with the traditional method of isolating malicious nodes, the PGSO algorithm integrated with trust value performs better, especially under a large scale of nodes.

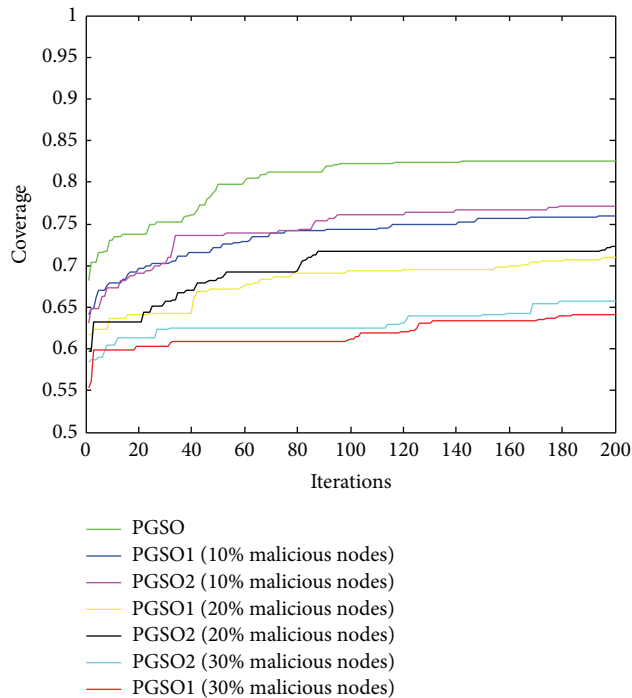


FIGURE 6: The process of convergence when the node suffers invasion.

The current problems focused on the following three aspects. First of all, there are many types of node attacks, for example, replay attacks and so on, so the trust factor in the trust value model needs to be improved. Secondly, when the node is judged to be a malicious one, more experiments and follow-up studies are needed in order to find a formula with more appropriate trust value and radius corresponding regulation. In addition, due to the energy limitation of mobile sensor network node, it is of great significance to extend the life coverage of the reliable network appropriately when considering the node trust values and energy consumption.

Conflict of Interests

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

Acknowledgments

This work was sponsored by the Natural Science Foundation of Hunan Province, China (13JJ3091, 14JJ3062), National Nature Science Foundation, China (61202462, 61300036), and the Fundamental Research Funds for the Central Universities, China.

References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam et al., "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [2] K. Nagarathna, Y. B. Kiran, J. D. Mallapur, and S. Hiremath, "Trust based secured routing in wireless multimedia sensor networks," in *Proceedings of the 4th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN '12)*, pp. 53–58, IEEE Press, Piscataway, NJ, USA, July 2012.
- [3] Y. Chen and J. Shu, "Wireless sensor networks security issues as smart materials systems," *Applied Mechanics and Materials*, vol. 63–64, pp. 497–501, 2011.
- [4] Z. Y. Tao and M. Hu, "Time synchronization algorithm based on hierarchical structure in wireless sensor network," *Journal of Computer Applications*, vol. 32, no. 6, pp. 1513–1515, 2012.
- [5] H. Gobjuka and Y. Breitbart, "Discovering network topology of large multi subnet ethernet networks," in *Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN '07)*, pp. 428–435, October 2007.
- [6] L. Qionglin, "Inclusion-exclusion principle and application," *China Science and Technology Information*, vol. 20, no. 8, pp. 58–59, 2012.
- [7] A. Francy Golda, S. Aridha, and D. Elakkiya, "Algorithmic agent for effective mobile robot navigation in an unknown environment," in *Proceedings of the International Conference on Intelligent Agent and Multi-Agent Systems (IAMA '09)*, pp. 1–4, Chennai, India, July 2009.
- [8] W. R. Pires, T. H. P. Figueiredo, H. C. Wong, and A. A. F. Loureiro, "Malicious node detection in wireless sensor networks," in *Proceedings of the 18th International Parallel and Distributed Processing Symposium*, 2004.
- [9] C. Wang, X. Y. Jia, and Q. Lin, "Trust based secure routing algorithm for wireless sensor networks," *Journal on Communications*, vol. 29, no. 11, pp. 105–112, 2008.
- [10] L. Zhang, L. Li, and C. Li, "A wireless sensor network routing algorithm based on subjective logic trusted," *Wuhan University of Technology*, vol. 33, no. 1, pp. 75–78, 2009.
- [11] Z. Cheng, Z. Ming-Zheng, and X. Jinsheng, "BTSR: A behavior-based safety data fusion and routing algorithm credible," *Journal of Computer Applications*, vol. 28, no. 11, pp. 2820–2823, 2008.
- [12] D. Shuhao and L. Xiaolon, "Reliable coverage algorithm in wireless sensor networks based on grid trust," *Application Research of Computers*, vol. 31, no. 1, pp. 253–256, 2014.
- [13] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, Oxford, UK, 1999.
- [14] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [15] L. Jianping, X. Li, and C. Minrong, "SFLA the Markov model and its convergence analysis," *Acta Electronica Sinica*, vol. 38, no. 12, pp. 2875–2880, 2010.
- [16] E. Sun, "A survey on clustering routing protocols based on PSO in WSN," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, no. 7, 2014.
- [17] J.-P. Luo, X. Li, and M.-R. Chen, "The Markov model of shuffled frog leaping algorithm and its convergence analysis," *Acta Electronica Sinica*, vol. 38, no. 12, pp. 2875–2880, 2010.
- [18] K. N. Krishnanand and D. Ghose, "Theoretical foundations for rendezvous of glowworm-inspired agent swarms at multiple locations," *Robotics and Autonomous Systems*, vol. 56, no. 7, pp. 549–569, 2008.

- [19] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization: a new method for optimizing multi-modal functions," *International Journal of Computational Intelligence Studies*, vol. 1, no. 1, pp. 93–119, 2009.
- [20] K. Huang and Y. Zhou, "Improved variation step adaptive GSO algorithm," *Computer Engineering*, vol. 38, no. 4, pp. 185–187, 2012.

Research Article

Accurately Identifying New QoS Violation Driven by High-Distributed Low-Rate Denial of Service Attacks Based on Multiple Observed Features

Jian Kang,^{1,2} Mei Yang,³ and Junyao Zhang⁴

¹Department of Computer Science & Technology, Jilin University, Changchun 130012, China

²Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China

³Department of Software Engineering, Jilin University, Changchun 130012, China

⁴Department of EECS, University of Central Florida, Orlando, FL 32816, USA

Correspondence should be addressed to Jian Kang; kj885788@gmail.com

Received 4 August 2014; Revised 24 November 2014; Accepted 8 December 2014

Academic Editor: Jun Zhang

Copyright © 2015 Jian Kang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose using multiple observed features of network traffic to identify new high-distributed low-rate quality of services (QoS) violation so that detection accuracy may be further improved. For the multiple observed features, we choose *F feature* in TCP packet header as a microscopic feature and, *P feature* and *D feature* of network traffic as macroscopic features. Based on these features, we establish *multistream fused hidden Markov model* (MF-HMM) to detect stealthy low-rate denial of service (LDoS) attacks hidden in legitimate network background traffic. In addition, the threshold value is dynamically adjusted by using Kaufman algorithm. Our experiments show that the additive effect of combining multiple features effectively reduces the false-positive rate. The average detection rate of MF-HMM results in a significant 23.39% and 44.64% improvement over typical power spectrum density (PSD) algorithm and nonparametric cumulative sum (CUSUM) algorithm.

1. Introduction

In recent years malicious quality of services (QoS) violation attacks have become one of the most serious security threats to the Internet. New QoS attacks are increasingly showing the trend of high-distributed low rate. In the literature, this kind of attacks has been called *shrew attacks* [1], *pulsing denial of service (DoS) attacks* [2], or *reduction of quality (RoQ) attacks* [3]. For simplicity, we call all of them *LDoS (low-rate denial of service) attacks* in the sequel.

LDoS attacks are stealthy, periodic, pulsing, and low rate in attack volume, very different from early flooding type of attacks. A traditional detection system against flooding attacks is based on traffic volume analysis method in the time domain. However, it almost has no effect on new LDoS attack [4]. This is because the average bandwidth consumption differs very little between normal and attack streams.

In this paper, we present a new approach to identify LDoS attacks by combining multiple observed features at the micro- and macrolevel. Multidimensional features are extremely valuable for describing slight changes of network properties and help us accurately differentiate attack flows. So our new approach can complement existing detection mechanisms based on one-dimensional feature and overcome the bottleneck of detection accuracy for LDoS violation.

In microscopic features, we calculate *weighted summation of flag bits* (WSFB) in TCP packet header to reflect the packet's internal slight change with and without LDoS attacks. Macroscopically, the best distinguishing characteristic between LDoS and normal flow is different periodicity in frequency domain [5]. Based on this fact, we choose *weighted average size of packet in queue* (WASPQ) in router as an observed sequence. Then, we convert the WASPQ sequence into frequency-domain spectrum using discrete

Fourier transform (DFT) and achieve the power spectrum density (PSD) of WASPQ as a macroscopic feature. Moreover, we calculate the *difference between request/response flows* (DRRF) as another macroscopic feature.

Based on above three-dimensional features, we develop a multistream fused hidden Markov model (MF-HMM) to detect LDoS violation hidden in legitimate TCP/IP traffic. In addition, we adjust the *decision threshold* value dynamically based on Kaufman algorithm for improving the detection accuracy. Notations, symbols, and abbreviations used in this paper are summarized in Notations section. Only brief definitions are given here; details are given in subsequent sections.

The rest of this paper is organized as follows. In Section 2, we present the related work. Section 3 describes MF-HMM, its advantages, and its training algorithm. Section 4 presents the overview of TF-HMM procedure and explains how to extract multiobserved features of network traffic to establish the corresponding component HMM of TF-HMM. Furthermore, we also introduce the threshold dynamic adjustment based on Kaufman algorithm. In Section 5, we compare our work with those of other researchers and discuss the training and recognition time of TF-HMM. Finally, we conclude our paper in Section 6.

2. Related Work

Some scholars studied the mathematical model of LDoS attacks. By simulating various LDoS attacks, they discussed the properties of LDoS attacks and gave some suggestions on further research. Maciá-Fernández et al. [6] summarized the behavior of LDoS and proposed a mathematical model for the LDoS attack. They also discussed the development trend and made some recommendations for building defense techniques against this attack. He et al. [7] presented theoretical analyses, modeling, and simulations of various LDoS attacks. And they discussed the difficulties of defending and current solutions. Zhu et al. [8] discussed the vulnerabilities of TCP and the principle of low-rate attacks. Moreover, the simulation of attacks was investigated, and the further direction of research is suggested.

Most current LDoS-related studies focus on using the frequency domain method to detect LDoS attack and have made clear progress. A research group [9] proposed an approach of detecting LDoS attack based on the model of small signal. Furthermore, in paper [10], they presented the method of multiple sampling averaging based on missing sampling (MSABMS) to detect LDoS attacks. An eigenvalue-estimating matrix was established to estimate the attack period after the detection of LDoS attacks. In addition, they also indicated a scheme [11] of detecting LDoS attack based on time window sampling in time domain and capturing the periodicity by statistic analysis in frequency domain. Zhang et al. [12] proposed a detection method, which is similar to that of Yu et al. [13]. In this method, the sum of the power spectrum is computed within 1-50 Hz, and the intersection of the two fitting curves is taken as the judging threshold. Luo and Chang [2] proposed a two-stage scheme to detect

LDoS attacks on a victim network. The first stage is a discrete wavelet transform (DWT) analysis of the network traffic. The second stage is to detect change points by using a non-parametric cumulative sum (CUSUM) algorithm. Liu [14] proposed an LDoS attack detection method by calculating the *Holder* based on binary discrete wavelet analysis. Shevtekar et al. [15] presented an approach of detecting the periodicity of attack flow based on autocorrelation of flow.

Some detection methods based on traditional traffic characteristics are proposed in recent years. These methods detect the LDoS attacks by searching and identifying the abnormal network traffic caused by the LDoS attacks. For example, the exponentially weighted moving average (EWMA) method was presented in papers [16, 17]. However, the EWMA algorithm may smooth not only the normal traffic but also the abnormal traffic. This will affect the detection accuracy for the LDoS attacks. Therefore, paper [18] proposed an adaptive EWMA method which used an adaptive weighting function instead of the constant weighting of EWMA algorithm. The adaptive EWMA can smooth the accidental error and retain the exceptional mutation. Thus, it is more efficient than EWMA method.

Unlike a popular deployment location of detection system, paper [19] proposed an adaptive detection method for LDoS attacks in *source-end* network. The method does not require the distribution assumption of the traffic samples. Moreover, they presented the automatic adjustment of the detection threshold according to the traffic conditions.

In particular, Xiang et al. [20] innovatively propose using two new information metrics to detect low-rate DDoS attacks by measuring the difference between legitimate traffic and attack traffic. The proposed *generalized entropy metric* and *information distance metric* outperform the existing popular approach as they can clearly enlarge the adjudication distance and then obtain the better detection sensitivity.

In summary, most researches use one-dimensional information of network traffic to establish algorithms for detecting LDoS attack. Though some algorithms are sophisticated, one-dimensional information is not enough to accurately differentiate stealthy LDoS attack hidden in legitimate traffic. Despite gratifying progress, the high false-positive rate is still a striking bottleneck.

3. Multistream Fused HMM

We first describe basic properties of multistream fused HMM and then give its mathematical description and training algorithm in detail.

3.1. Basic Properties. To accurately identify stealthy LDoS violation hidden in legitimate network traffic, the combination of multiobserved features is considered in our scheme by using multistream fused HMM [21]. According to the maximum entropy principle and the maximum mutual information (MMI) criterion, MF-HMM constructs a new structure linking multiple HMMs. MF-HMM is the generalization of two-stream fused HMM [22].

The main advantages of MF-HMM are as follows.

- (1) Every observation feature can be modeled by a component HMM, so the performance of every feature can be analyzed individually. The set of features can be modified according to the performance analysis.
- (2) Compared with other existing model fusion methods (e.g., CHMM [23], MHMM [24], etc.), MF-HMM reaches a better balance between model complexity and performance.
- (3) MF-HMM has stronger robustness. If one component HMM fails due to some reason, the other component HMM can still work. Thus, the final result is still a valuable reference for the recognition judgment.

3.2. Mathematical Description. HMM is the basis of MF-HMM. In brief, we only discuss MF-HMM, and paper [25] discussed the HMM definition and relevant algorithms in detail. The mathematical symbols in this paper are consistent with the standard HMM description symbol.

Let $\{O^{(i)}, i = 1, \dots, n\}$ represent n tightly coupled observing sequences. Assume that $\{O^{(i)}, i = 1, \dots, n\}$ can be modeled by n corresponding HMMs with hidden states $\{Q^{(i)}, i = 1, \dots, n\}$. In MF-HMM, an optimal solution for $p(O^{(1)}; O^{(2)}; \dots; O^{(n)})$ is given according to the maximum entropy principle and the maximum mutual information criterion $\hat{p}(O^{(1)}; O^{(2)}; \dots; O^{(n)})$.

In order to calculate $\hat{p}(O^{(1)}; O^{(2)}; \dots; O^{(n)})$, firstly we need to calculate every component $\hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)})$; here $i = 1, 2, \dots, n$. The i th $\hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)})$ can be given through

$$\begin{aligned} \hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)}) &= p(O^{(1)}) p(O^{(2)}) \dots p(O^{(n)}) \\ &\quad \cdot \frac{p(Q^{(i)}, O^{(1)}, \dots, O^{(i-1)}, O^{(i+1)}, \dots, O^{(n)})}{p(Q^{(i)}) p(O^{(1)}) \dots p(O^{(i-1)}) p(O^{(i+1)}) \dots p(O^{(n)})} \\ &= p(O^{(i)}) p(O^{(1)}, \dots, O^{(i-1)}, O^{(i+1)}, \dots, O^{(n)} | Q^{(i)}). \end{aligned} \quad (1)$$

And assume

$$\begin{aligned} p(O^{(1)}, \dots, O^{(i-1)}, O^{(i+1)}, \dots, O^{(n)} | Q^{(i)}) \\ = \prod_{j \neq i, j=1}^n p(O^{(j)} | Q^{(i)}). \end{aligned} \quad (2)$$

It has a good record in recognizing and detecting LDoS attacks, though the conditional independence assumption is always violated in practice. The success is because of the small number of parameters to be estimated in assumption. Without this assumption, some complicated algorithms require more training data and are more susceptible to local maximum during parameter estimation.

So, the estimate of $\hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)})$ can be given by

$$\hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)}) = p(O^{(i)}) \prod_{j \neq i, j=1}^n p(O^{(j)} | Q^{(i)}). \quad (3)$$

There are different expressions to different i . To our three-stream fused HMM (TF-HMM), (3) corresponds to (4a), (4b), and (4c) as follows;

$$\begin{aligned} \hat{p}^{(1)}(O^{(1)}; O^{(2)}; O^{(3)}) \\ = p(O^{(1)}) p(O^{(2)} | Q^{(1)}) p(O^{(3)} | Q^{(1)}) \end{aligned} \quad (4a)$$

$$\begin{aligned} \hat{p}^{(2)}(O^{(1)}; O^{(2)}; O^{(3)}) \\ = p(O^{(2)}) p(O^{(1)} | Q^{(2)}) p(O^{(3)} | Q^{(2)}) \end{aligned} \quad (4b)$$

$$\begin{aligned} \hat{p}^{(3)}(O^{(1)}; O^{(2)}; O^{(3)}) \\ = p(O^{(3)}) p(O^{(1)} | Q^{(3)}) p(O^{(2)} | Q^{(3)}). \end{aligned} \quad (4c)$$

In practice, if the n component HMMs have different reliabilities, they may be combined by different weights to get a better result:

$$\hat{p}(O^{(1)}; O^{(2)}; \dots; O^{(n)}) = \sum_{i=1}^n \lambda^{(i)} \hat{p}^{(i)}(O^{(1)}; O^{(2)}; \dots; O^{(n)}). \quad (5)$$

Here, $\sum_{i=1}^n \lambda^{(i)} = 1$.

3.3. Training Algorithm. The training algorithm of MF-HMM is a three-step process.

- (1) n component HMMs are trained independently by representative algorithm, such as Baum-Welch algorithm, segmented K-means algorithm, or hybrid method EM algorithm.
- (2) The best hidden state sequences of the component HMMs are estimated by the Viterbi algorithm.
- (3) Calculate the coupling parameters between the n HMMs.

To our three-stream fused HMM, step (1) is to calculate (6a), (6b), and (6c):

$$\hat{\Pi}^{(1)}, \hat{A}^{(1)}, \hat{B}^{(1)} = \arg \max_{\Pi^{(1)}, A^{(1)}, B^{(1)}} (\log p(O^{(1)})) \quad (6a)$$

$$\hat{\Pi}^{(2)}, \hat{A}^{(2)}, \hat{B}^{(2)} = \arg \max_{\Pi^{(2)}, A^{(2)}, B^{(2)}} (\log p(O^{(2)})) \quad (6b)$$

$$\hat{\Pi}^{(3)}, \hat{A}^{(3)}, \hat{B}^{(3)} = \arg \max_{\Pi^{(3)}, A^{(3)}, B^{(3)}} (\log p(O^{(3)})). \quad (6c)$$

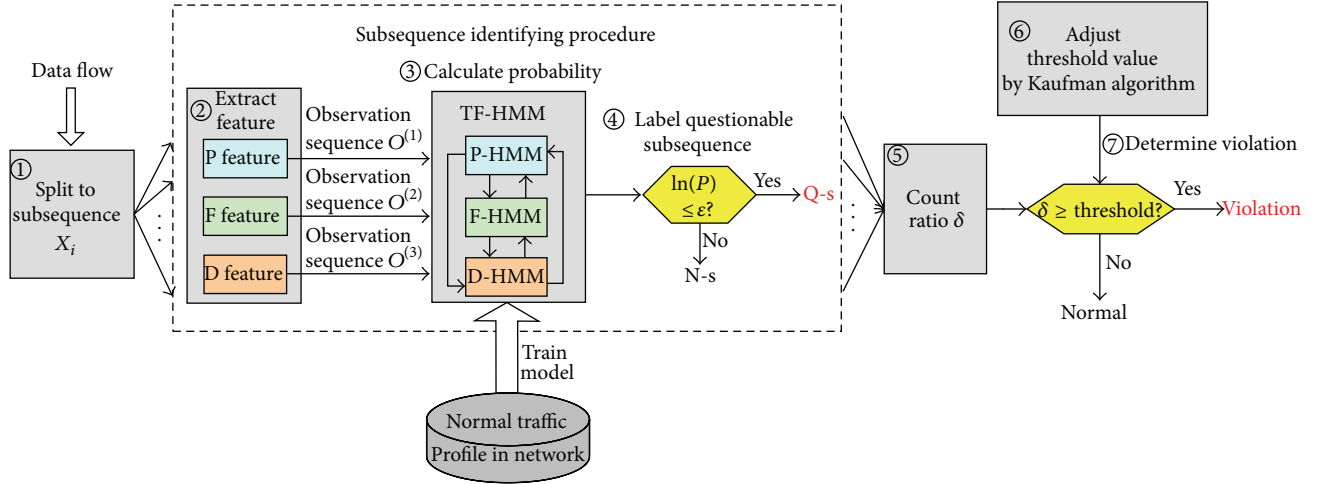


FIGURE 1: Procedure of TF-HMM.

Step (2) is to calculate (7a), (7b), and (7c):

$$\widehat{Q}^{(1)} = \arg \max_{Q^{(1)}} (\log p(O^{(1)}, Q^{(1)})) \quad (7a)$$

$$\widehat{Q}^{(2)} = \arg \max_{Q^{(2)}} (\log p(O^{(2)}, Q^{(2)})) \quad (7b)$$

$$\widehat{Q}^{(3)} = \arg \max_{Q^{(3)}} (\log p(O^{(3)}, Q^{(3)})). \quad (7c)$$

Step (3) is to estimate the coupling parameters between HMM1, HMM2, and HMM3:

$$\widehat{B}^{(1,2)} = \arg \max_{B^{(1,2)}} p(O^{(2)} | \widehat{Q}^{(1)}) \quad (8a)$$

$$\widehat{B}^{(1,3)} = \arg \max_{B^{(1,3)}} p(O^{(3)} | \widehat{Q}^{(1)}) \quad (8b)$$

$$\widehat{B}^{(2,1)} = \arg \max_{B^{(2,1)}} p(O^{(1)} | \widehat{Q}^{(2)}) \quad (8c)$$

$$\widehat{B}^{(2,3)} = \arg \max_{B^{(2,3)}} p(O^{(3)} | \widehat{Q}^{(2)}) \quad (8d)$$

$$\widehat{B}^{(3,1)} = \arg \max_{B^{(3,1)}} p(O^{(1)} | \widehat{Q}^{(3)}) \quad (8e)$$

$$\widehat{B}^{(3,2)} = \arg \max_{B^{(3,2)}} p(O^{(2)} | \widehat{Q}^{(3)}). \quad (8f)$$

4. Identifying LDoS Violation Using TH-HMM

In this section, we first present the procedure of identifying LDoS violation by using TF-HMM. Then, we explain how to establish three-component HMMs of TF-HMM, including F-HMM, P-HMM, and D-HMM. At last, we introduce the threshold dynamic adjustment based on Kaufman algorithm.

4.1. Procedure Overview. In order to make it easier to understand, we firstly introduce the procedure of TH-HMM, as illustrated in Figure 1.

(1) *Split into Subsequence.* Let the length of the detected sequence be L . Split the detected sequence with a k length

splitting window, so the set of these subsequences is $\{X_i\}$; here, $1 \leq i \leq L/k$.

(2) *Extract Three Observed Features.* Extract F feature, P feature, and D feature, and then form the three-dimensional observation state sequence.

(3) *Calculate the Output Probability.* Input state sequences to TF-HMM, and calculate the output probability $\ln \widehat{p}(O^{(1)}; O^{(2)}; O^{(3)})$ of every subsequence, denoted by $\ln(P)$.

(4) *Label a Questionable Subsequence.* If $\ln(P)$ is less than the threshold ϵ , it is labeled as a questionable subsequence (Q-s); otherwise it is marked as a normal subsequence (N-s).

(5) *Count the Ratio of Questionable Subsequence.* After computing and labeling all subsequences, count the ratio δ according to

$$\delta = \frac{\text{the number of questionable subsequences}}{\text{the total of all subsequences}}. \quad (9)$$

(6) *Adjust Threshold Value by Kaufman Algorithm.* During the detection system run, the threshold value will be adjusted by using Kaufman algorithm. In practice, the average detection rate of TF-HMM has been effectively improved with it.

(7) *Determine the Violation.* At last, compare δ with the decision threshold value *threshold*: if $\delta > \text{threshold}$, it is determined as LDoS violations; else, there is no violations.

4.2. Establishing Three-Component HMMs. In order to apply TF-HMM, we extract multiobserved features of network traffic, including WSFB feature, PSD of WASPQ feature, and DRRF feature. They constitute three-dimensional observation state sequence. Each sequence is modeled by a component HMM. Three-component HMMs together make up TF-HMM.

TABLE 1: Weight of different flag bits in TCP header.

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N
2^5	2^4	2^3	2^2	2^1	2^0

4.2.1. F-HMM. In order to reduce network QoS, spoofed TCP/IP packets must be used. In microscopic view, attackers usually use random number to fill internal attribute fields of the forged packet, resulting in vast differences with the real data packet. We choose flag bits in TCP packet header as a microscopic feature to describe a slight internal change of packet attribute fields.

To enlarge differences of flag bits between the forged packets with the real ones, we define different weights to different flag bits [26], as in Table 1.

Next, we achieve the *weighted summation of flag bits* (WSFB) by using

$$O_{\text{wsfb}} = 2^5 \times \text{URG} + 2^4 \times \text{ACK} + 2^3 \times \text{PSH} + 2^2 \times \text{RST} + 2^1 \times \text{SYN} + 2^0 \times \text{FIN}. \quad (10)$$

So we can construct a component HMM based on the observing sequence of WSFB; simply mark it as F-HMM.

4.2.2. P-HMM. Paper [27] indicates that attack data packet occupies a certain proportion in router buffer queue at LDoS attack, and the greater the damage is, the higher the proportion is. At the same time, paper [28, 29] concludes that attackers must use the data packet as short as possible to achieve a good attack effect, which results in an obvious decrease of the average size of packets in buffer queue under attacks than under normal conditions. We introduce the *weighted average size of packet in queue* (WASPQ) to describe this periodicity change in macroscopic view.

Let the number of packets in queue when at sampling time t be N_t and let each size of packet be S_i , $i \in [1, N]$. In order to highlight the characteristic that the shorter the packet, the more important, we introduce weight γ_i , $\gamma_i \in [0, 1]$, and calculate the WASPQ value S_{WASPQ} as follows:

$$S_{\text{WASPQ}} = \frac{\sum_{i=1}^{N_t} \gamma_i S_i}{N_t}. \quad (11)$$

In order to depict inherent periodic feature of LDoS attack, we take S_{WASPQ} as the discrete signal series and sample it with a period of 0.1 sec. The change of S_{WASPQ} value with and without attacks is modeled by a random process: $\{s(t), t = n\Delta, n \in N\}$, where Δ is a constant time interval, which we assume 0.1 sec, and N is a set of positive integers, and, at each time point t , $s(t)$ is a random variable, representing the total number of S_{WASPQ} in $(t - \Delta, t]$.

To study the periodicity embedded in the $s(t)$ sequence, we use its autocorrelation function in discrete time as follows:

$$R_{xx}(m) = \frac{1}{N-m} \sum_{n=0}^{N-m+1} [s(n) s(n+m)]. \quad (12)$$

The $R_{xx}(m)$ captures the correlation of the $s(t)$ sequence and itself at interval m . If there is any periodicity existing, autocorrelation function is capable of finding it.

To figure out the periodicity embedded in the $s(t)$ sequence, we convert the autocorrelation time series by discrete Fourier transform (DFT) to generate the power spectrum density (PSD) as follows:

$$\text{PSD}(f) = \text{DFT}(R_{xx}(m), f) = \frac{1}{N} |X[f]|^2, \quad (13)$$

where $X[f] = \sum_{n=0}^{N-1} R_{xx}(m) \exp(-j2\pi fn/N)$ is the N -point DFT, $f = 0, 1, 2, \dots, N-1$.

We note that we use the standard periodogram rather than Welch's method of averaged periodogram [30]. This is because in our work we are interested in the detection and estimation of a single periodic feature, which is better achieved using the standard periodogram as discussed in [31].

Therefore, we can get the component HMM based on the PSD of WASPQ feature, simply referred to as P-HMM.

4.2.3. D-HMM. In a normal TCP session of two-way communications, the request flow is limited by the response flow [32]. In the macroscopic view, the difference value between them should remain relatively stable normally. In case of LDoS attacks, a huge number of forgery request packets will lead to a sharp increase of the difference. Therefore, we introduce the *difference between request/response flows* (DRRF) to represent the difference change.

Let the sequence $d[i]$ be the difference value between request flow and response flow;

$$d[i] = f[i] - g[i] \quad (i = 0, 1 \dots), \quad (14)$$

where $f[i]$ is a request flow and $g[i]$ is a response flow.

Usually, $d[i]$ is closely related to the network size, the number of hosts, and the sampling time. In order to counteract the influence of them, we convert it as follows:

$$K[i] = \begin{cases} 0 & i = 0 \\ \alpha * K[i-1] + (1-\alpha) * g[i] & i = 1, 2, \dots \end{cases} \quad (15)$$

$$\text{DRRF}[i] = \frac{d[i]}{K[i]} \quad (i = 1, 2 \dots). \quad (16)$$

In formula (15), $K[i]$ could be expressed as a recurrence relation of $g[i]$, where α is a custom constant, $\alpha \in [0..1]$. Thus, by using formula (16), we can get $\text{DRRF}[i]$, which will not be impacted by factors mentioned above. Instead, it is simply about current network traffic. We choose $\text{DRRF}[i]$ as another macroscopic feature to indicate the overall change of two-way communications caused by LDoS attacks.

So we can establish a component HMM based on DRRF feature, simply referred to as D-HMM.

4.3. Adjusting Threshold Dynamic. Enlightened by load-shedding method and Kaufman algorithm [33], we adjust the *threshold* value dynamically for improving the detection precision.

Let the $\Gamma[i]$ denote the mapping variable of the system effective payload and our algorithm threshold in the $(i + 1)$ th time span. Define $\Gamma[0] = 1$. The range of $\Gamma[i]$ values is in $[\Gamma[\min], 1]$, where $\Gamma[\min]$ is a rather small but not 0 constant. This is because if $\Gamma[\min]$ is 0, all data flows are not allowed to pass through it. Hypothesize that, right at the i th time over, the actual payload in the system is $\rho[i]$, and $\rho[\max]$ is the maximum number of payload, so we get $\varphi[i] = \rho[\max]/\rho[i]$. $\Gamma[i]$ could be presented in a recursive way as follows:

$$\Gamma[i] = \Gamma[i - 1] * \varphi[i]. \quad (17)$$

And since $\Gamma[i] \in [\Gamma[\min], 1]$, we can get the final equation of $\Gamma[i]$; that is,

$$\Gamma[i] = \max \left\{ \min \left\{ \Gamma[0] * \prod_{j=1}^i \varphi[j], 1 \right\}, \Gamma[\min] \right\}, \quad (18)$$

where $i = 1, \dots, n$.

In this way, threshold value could be computed out by $\Gamma[i]$.

5. Experiments and Performance Results

In this section, we firstly introduce experimental environment setup. Then, we compare the normal flow with the attack one in aspect of the periodicity of WASPQ and the output of TF-HMM. Based on the comparisons, we validate the sufficient sensitivity of TF-HMM. Finally, we evaluate the performance results of TF-HMM in terms of detection rate, false-positive rate, average detection rate, training time, and recognition time.

5.1. Experimental Environment Setup. Data acquisition in real LDoS attacks is very difficult. Enlightened by papers [34–36], we construct experimental data by fusing controlled attack flows into real network background traffic.

To generate attack data, we have built a controlled experimental platform. 60 VMware hosts based on Windows XP system are chosen as user hosts. The collector and analyzer of network traffic are installed at Ubuntu 12.04 with Quad core 2.4 GHz CPU and 4 G RAM. We install Zombie tools at part of user hosts as bots. The controlled LDoS attack is launched by these bots, and then our experimental attack data could be achieved.

Accordingly, we choose a day's network traffic of a primary node in CERNET backbone networks as our experimental background traffic. There are 305985 records in the time window of 10 minutes. After the preprocessing, the background data contains 19877 hosts. Then, we fuse the attack data into the background traffic to evaluate TF-HMM performance.

5.2. Periodicity Analysis of WASPQ in P-HMM. The most obvious contrast between LDoS and normal flow is different periodicity in frequency domain. We firstly compare the normal WASPQ value with the attack WASPQ value.

As illustrated in Figure 2(a), in normal condition, the value of WASPQ is relatively high, almost 1100, because of the

small proportion of short data packet in cache queue. In case of LDoS attacks, attackers use massive number of very short data packet to launch suddenly, and the value of WASPQ declines abruptly as shown in Figure 2(b), from about 1100 to 50. This is due to the fact that we use the weighted approach and highlight the importance of short packet in WASPQ calculation. We go on to draw the according periodograms of Figures 2(a) and 2(b). As you can see in Figure 2(d), in case of LDoS attacks, the change of WASPQ has obvious periodicity, while normal flow has none in Figure 2(c).

Next, we draw the corresponding PSD of WASPQ, as shown in Figure 3. We can see that there is a very wide frequency band in normal condition, but when attacking, the PSD value is almost below 51.5 Hz, and there is no distribution in higher frequency bands. We calculate the cumulative traffic spectrum (CTS) [5] of PSD, as shown in Figure 4. 98.65% power of attack flow distributes under 51.5 Hz. Relatively, 39.44% power of normal flow is lower than 51.5 Hz. The huge difference can make P-HMM the better detection sensitivity.

5.3. Comparison Output of TF-HMM in Normal and in Attack.

In order to validate the sensibility of TF-HMM, we extract 30 seconds normal flow fragment firstly. Secondly, we extract 30 seconds fragment of LDoS violation and overlap them to one time axis. As shown in Figure 5, in normal, the value fluctuate in the range of $-40 \sim -984$, while, under attacking, the peak value could reach $2.4 \sim 55$ times more than normal value, or even larger. The red curve in Figure 5 obviously shows the 5 impulse low-rate violations, so it can be seen that TF-HMM has enough detection sensitivity to identify LDoS attacks hidden in legitimate network traffic.

5.4. Detection Rate and False-Positive Rate. In this section, we compare TF-HMM with representative nonparametric CUSUM algorithm [14] and PSD method [12] in detail. We focus on the detection accuracy and false positives of three algorithms in different network traffic. In order to evaluate impartially, various network traffics are employed in the following experiments, including different network utilization rates and attack intensions with or without legitimate periodicity flows. For simplicity, we call legitimate periodicity flows *the interference* in the sequel.

First, define detection rate R_d as

$$R_d = \frac{N_c}{N_r}. \quad (19)$$

Here, N_c is the number of attacks which have been detected correctly. N_r is the number of real attacks existing.

Next, define false-positive rate R_{fp} as

$$R_{fp} = \frac{N_a - N_c}{N_a}, \quad (20)$$

where N_a is the number of alarms by the detection algorithm and the difference between N_a and N_c is the number of false positives.

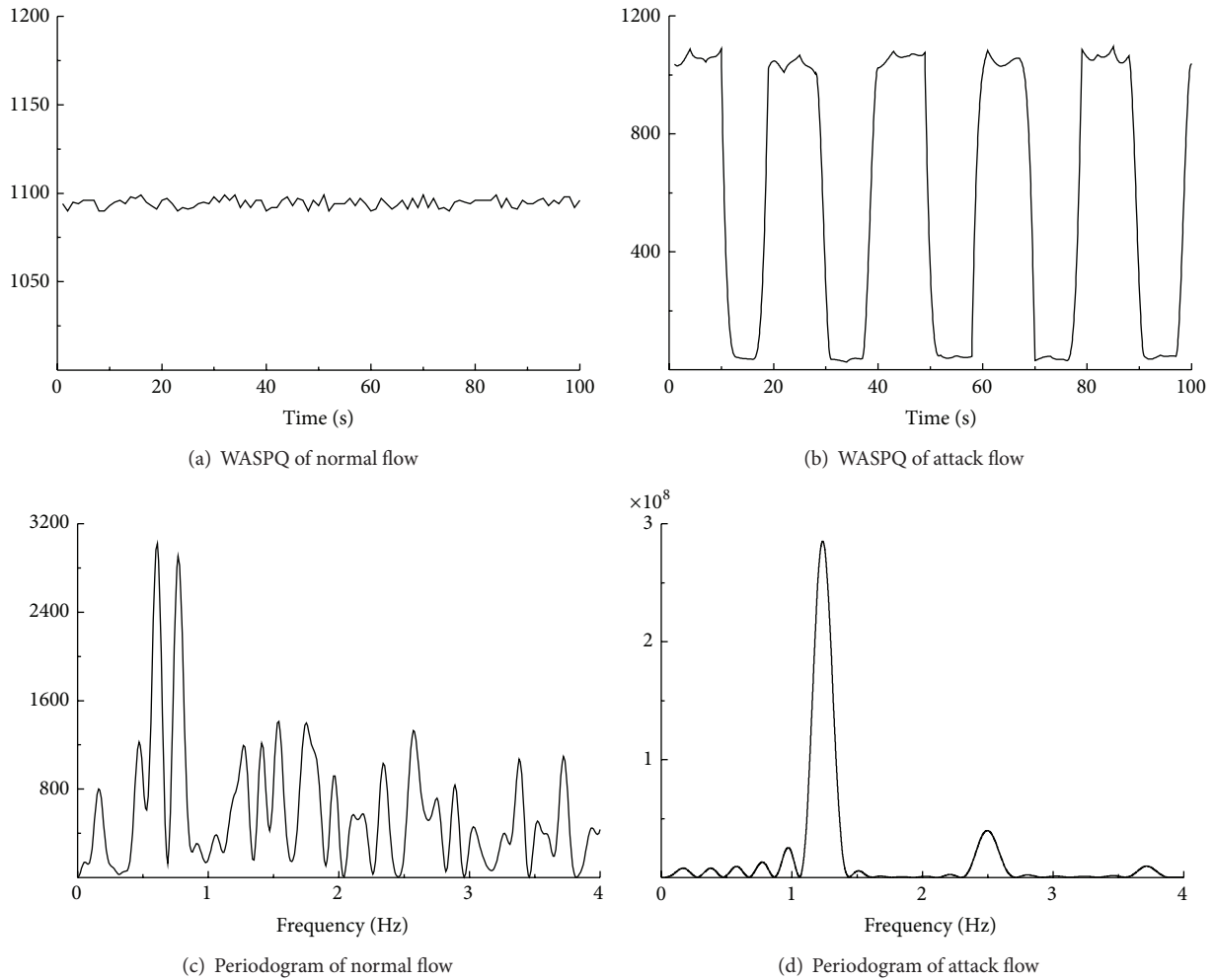


FIGURE 2: Comparison of WASPQ value and periodogram in normal and in attack flow.

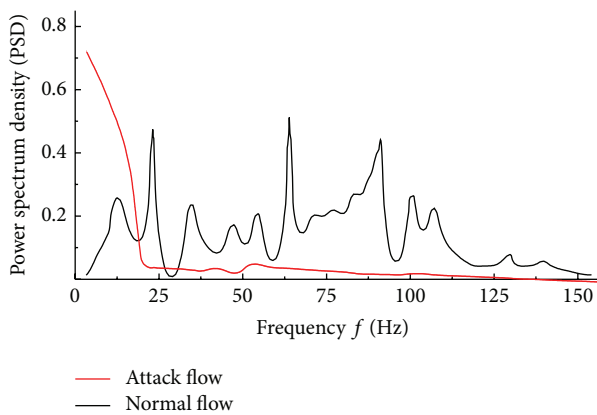


FIGURE 3: Comparison of normalized PSD of WASPQ in normal and in attack flow.

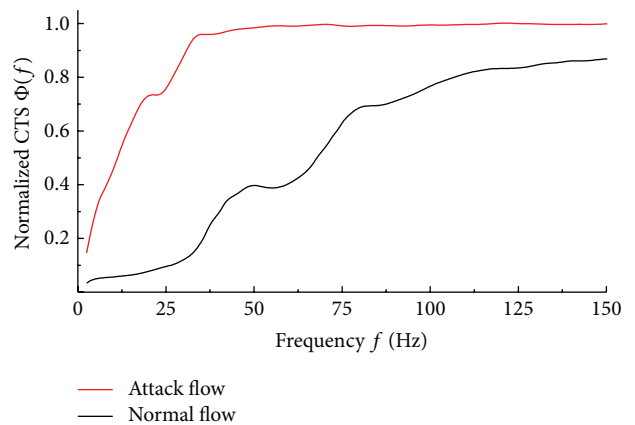


FIGURE 4: Comparison of CTS of WASPQ in normal and in attack flow.

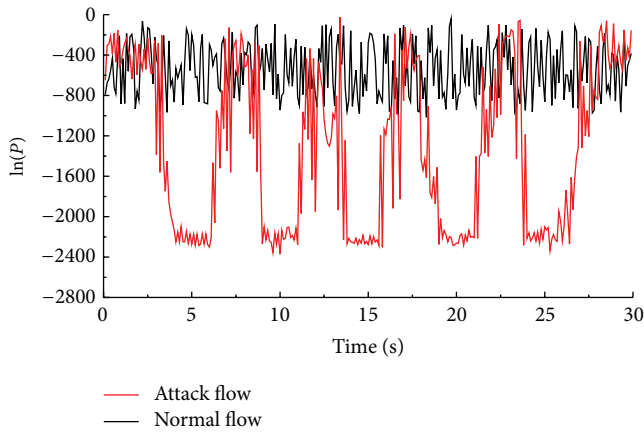
The experiment results are shown as in Table 2.

(1) Without Attacks and without the Interference (See No. 1 Group). There are no periodicity flows, so periodicity-based

algorithms (PSD and TF-HMM's P-HMM) give no false positives. However, CUSUM algorithm shows 3 false positives. This is because it is based on traffic volume accumulated

TABLE 2: Detection rate and false-positive rate comparison of 3 algorithms in different network traffic.

No.	Algorithm	N_a	N_c	N_r	Network utilization (%)	Interference
1	CUSUM	3	0		31.13	Without
	PSD	0	0	0		
	TF-HMM	0	0			
2	CUSUM	45	0		83.61	With
	PSD	18	0	0		
	TF-HMM	0	0			
3	CUSUM	23	1		33.15	Without
	PSD	2	2	2		
	TF-HMM	2	2			
4	CUSUM	45	19		47.23	With
	PSD	40	25	30		
	TF-HMM	33	29			
5	CUSUM	768	178		83.36	With
	PSD	588	250	300		
	TF-HMM	323	279			

FIGURE 5: Comparison of output $\ln(P)$ of TF-HMM in normal and in attack flow.

method in the time domain, having no analysis capabilities of frequency domain.

(2) *Without Attacks and with the Interference (See No. 2 Group)*. When injecting the interference flows and increasing utilization rate of network, false positives start appearing in the PSD algorithm but not in TF-HMM. This is due to the fact that the PSD cannot differentiate between the periodicity of the interference flows and one of pulse attacks, just capturing the periodicity. While TF-HMM's P-HMM can not only find the periodicity of flows but also analyze the WASPQ changes caused by LDoS attacks, it helps TF-HMM make an accurate distinction between legitimate periodicity flows and LDoS pulse flows.

TABLE 3: Average detection rate comparison of 3 algorithms.

CUSUM	PSD	TF-HMM
48.14%	69.39%	92.78%

(3) *With Attacks and without the Interference (See No. 3 Group)*. Without the interference, the PSD and TF-HMM can identify exactly 2 times attacks hidden in background traffic based on the obvious periodicity of pulse attacks and show no false positives. But CUSUM still remains relatively high false positives because it is not a learning-oriented algorithm and is not also a frequency-domain-based one.

(4) *With Attacks and with the Interference (See No. 4 and No. 5 Group)*. In No. 4 group, the result from TF-HMM is closer to REAL than other algorithms. Its R_{fp} is 12.12%. Conversely, the R_{fp} of CUSUM reaches up to 57.78%; the R_{fp} of PSD is 37.50%. With a growing intension of attacks and interferences, the R_{fp} of other two methods will be even higher.

In No. 5 group, we increased both of the attack intension and network utilization rate, and the advantages of TF-HMM based on multiple observed features becomes apparent. The R_{fp} of CUSUM is 76.82% and the R_{fp} of PSD is 57.48%, while its R_{fp} is 13.62%, far less than other algorithms. The reason for such low the R_{fp} of TF-HMM is that the two components of F-HMM and P-HMM play an important role.

When massive packets of legitimate periodicity flows and pulse attack flows arrive at the router, the PSD algorithm cannot accurately differentiate between them because it only uses the number of packet arrivals as a single periodic feature to find the periodicity in data sequence. Rather, the P-HMM can identify them because of WASPQ value abnormal decrease by pulse attacks (As illustrated in Figure 2(b)). Furthermore, the F-HMM can detect the packet's internal attribute fields that have been tampered with, because spoofed packets in pulse attacks result in abnormal fluctuations of WSFB.

The additive effect of combining multidimensional features starts to dominate, so we see a lower false-positive rate of TF-HMM. These provide some of the advantages of detection accuracy in TF-HMM not only with the higher detection rate, but also with the lower false-positive rate.

5.5. *Average Detection Rate*. In order to evaluate three detection approaches objectively, we varied attack intension, network utilization rate, sampling time, and the interference. Thus, there are obvious differences between every two groups. From the 100 groups of data gained, we calculated their average detection rate as presented in Table 3.

In Table 3, the average detection rate of PSD is obviously higher than CUSUM algorithm because it takes into account the inherent periodicity of LDoS violation. But the false positive rate is not still reduced to a reasonably low level; it limits the improvement of the detection accuracy. In contrast, since TF-HMM combines multiobserved features, its average detection rate reaches 92.78%, which is 1.93 times over CUSUM and 23.39% over PSD. It efficiently overcomes the bottleneck of limiting further increases in detection accuracy.

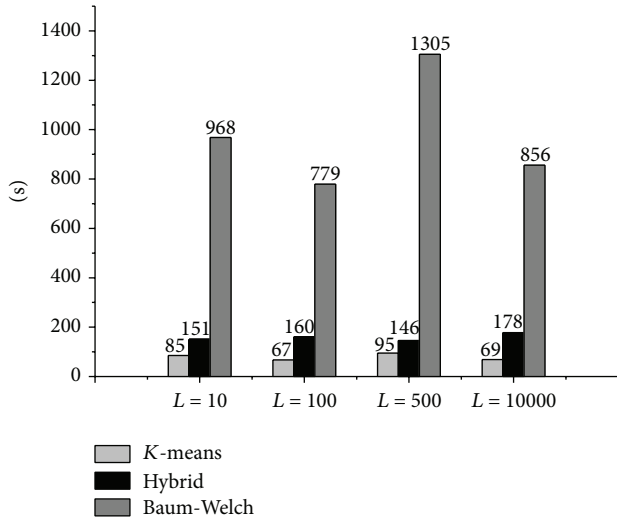


FIGURE 6: Influence of training time from training algorithm and segment length.

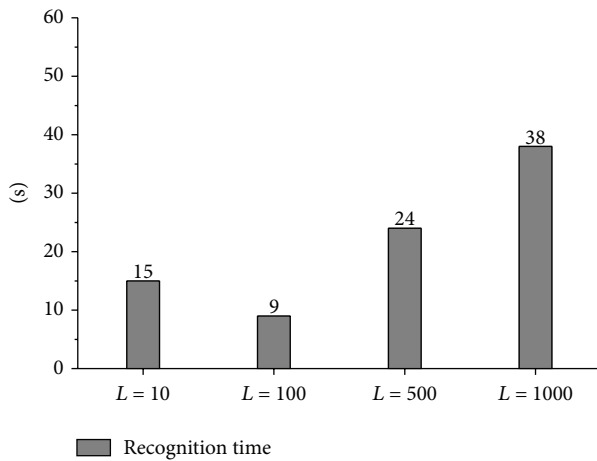


FIGURE 7: Influence of recognition time from segment length.

5.6. Training Time and Recognition Time. The time complexity of algorithms is vital to fast detection and response to QoS violation. Relevant experiments on training time and recognition time of the TF-HMM are sketched as in Figures 6 and 7.

As shown in Figure 6, the most time-consuming one is Baum-Welch algorithm; it is about 5 to 10 times of the other two algorithms; the second one is hybrid algorithm and then K-means algorithm. Furthermore, Baum-Welch algorithm is most sensitive to the length of segment. For example, using the same training sequence, the training time of $L = 500$ is 1.68 times more than the one of $L = 100$. But K-means and hybrid algorithms are insensitive to the length of segment.

And yet the recognition time of TF-HMM is short as shown in Figure 7. It is suitable for fast detection and responses to malicious QoS violations. Our ultimate goal is to achieve automated intrusion detection and responses in real time.

6. Conclusions

Current new LDoS violations are more and more characterized by high-distributed low rate. It is very difficult that fast detection and responses to stealthy LDoS streams are hidden in massive legitimate network traffic. The high false-positive rate is still the most striking bottleneck.

To overcome the bottleneck, our research contributions are summarized below in three technical aspects.

(1) *Combining Multidimensional Features.* Multiple micro- and macrofeatures, including WSFB, WASPQ, and DRRF, are combined together by using MF-HMM. The additive effects of combining multidimensional features make encouraging results on high detection rate with low false-positive rate.

(2) *Synthesizing Methods in Frequency Domain and in Time Domain.* Leveraging PSD analysis in the component P-HMM, we capture and identify the periodicity of LDoS pulse attacks in frequency domain. Furthermore, we calculate WSFB and DRRF feature in time domain by the components of F-HMM and D-HMM. These components make the accurate matching in detecting LDoS attacks at traffic streaming level.

(3) *Adjusting Threshold Value Dynamically.* Enlightened by load-shedding method and Kaufman algorithm, we adjust the threshold value dynamically to further reduce the false-positive rate.

For continued effort, we aim to improve the detection accuracy in complicated network traffic and ultimately to a fully automated process of detection and responses to LDoS attacks in real time.

Notations

CTS:	Cumulative traffic spectrum
CUSUM:	Cumulative Sum
D feature:	DRRF feature
DDoS:	Distributed denial of service
DFT:	Discrete Fourier transform
D-HMM:	The component HMM based on D feature
DoS:	Denial of service
DR:	Detection rate
DRRF:	Difference between request/response flows
DWT:	Discrete wavelet transform
F feature:	WSFB feature
F-HMM:	The component HMM based on F feature
LDoS:	Low-rate denial of Service
$\ln(P)$:	The output probability of TF-HMM
MF-HMM:	Multistream fused hidden Markov model
N-s:	Normal subsequence
P feature:	PSD of WASPQ feature
P-HMM:	The component HMM based on P feature
PSD:	Power spectrum density
QoS:	Quality of services
Q-s:	Questionable subsequence
RoQ:	Reduction of quality
TF-HMM:	Three-stream fused hidden Markov model
WASPQ:	Weighted average size of packet in queue
WSFB:	Weighted summation of flag bits.

Conflict of Interests

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

Acknowledgments

The authors would like to acknowledge the support of this work by National Natural Science Foundation of China (Grants nos. 60703023, 90204014) and Technology Development Plan of Jilin Province of China (Grant no. 20090110).

References

- [1] A. Kuzmanovic and E. W. Knightly, "Low-rate TCP-targeted denial of service attacks: the shrew vs. the mice and elephants," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Karlsruhe, Germany, 2003.
- [2] X. Luo and R. K. C. Chang, "On a new class of pulsing denial-of-service attacks and the defense," in *Proceedings of the Network and Distributed System Security Symposium (NDSS '05)*, San Diego, Calif, USA, 2005.
- [3] M. Guirguis, A. Bestavros, and I. Matta, "Exploiting the transients of adaptation for RoQ attacks on internet resources," in *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP '04)*, pp. 184–195, Berlin, Germany, October 2004.
- [4] M. Guirguis, A. Bestavros, I. Matta, and Y. Zhang, "Reduction of Quality (RoQ) attacks on Internet end-systems," in *Proceedings of the IEEE International Conference on Computer Communication (INFOCOM '05)*, pp. 1362–1372, Miami, Fla, USA, March 2005.
- [5] Y. Chen and K. Hwang, "Collaborative detection and filtering of shrew DDoS attacks using spectral analysis," *Journal of Parallel and Distributed Computing*, vol. 66, no. 9, pp. 1137–1151, 2006.
- [6] G. Maciá-Fernández, J. E. Díaz-Verdejo, and P. García-Teodoro, "Mathematical model for low-rate dos attacks against application servers," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 519–529, 2009.
- [7] Y. X. He, T. Liu, Q. Cao et al., "A survey of low-rate denial-of-service attacks," *Journal of Frontiers of Computer Science & Technology*, vol. 2, no. 1, pp. 1–17, 2008.
- [8] Q. Zhu, Z. Yizhi, and X. Chuiyi, "Research and survey of low-rate denial of service attacks," in *Proceedings of the 13th International Conference on Advanced Communication Technology: Smart Service Innovation through Mobile Interactivity (ICACT '11)*, pp. 1195–1198, Gangwon-Do, Republic of Korea, February 2011.
- [9] Z. J. Wu and B. S. Pei, "The detection of LDoS attack based on the model of small signal," *Acta Electronica Sinica*, vol. 39, no. 6, pp. 1456–1460, 2011.
- [10] W. Zhi-Jun, Z. Hai-Tao, W. Ming-Hua, and P. Bao-Song, "MSABMS-based approach of detecting LDoS attack," *Computers & Security*, vol. 31, no. 4, pp. 402–417, 2012.
- [11] Z.-J. Wu, H.-L. Zeng, and M. Yue, "Approach of detecting LDoS attack based on time window statistic," *Journal on Communications*, vol. 31, no. 12, pp. 55–62, 2010.
- [12] C.-W. Zhang, J.-P. Yin, Z.-P. Cai, and W.-F. Chen, "RRED: robust RED algorithm to counter low-rate denial-of-service attacks," *IEEE Communications Letters*, vol. 14, no. 5, pp. 489–491, 2010.
- [13] C. Yu, H. Kai, and Y.-K. Kwok, "Collaborative defense against periodic shrew DDoS attacks in frequency domain," *ACM Transactions on Information and System Security*, pp. 2–27, 2005.
- [14] D. Liu, *Research on LDoS attack in soft-switch network [M.S. thesis]*, Communication and Information Engineering, 2013.
- [15] A. Shevtekar, K. Anantharam, and N. Ansari, "Low rate TCP denial-of-service attack detection at edge routers," *IEEE Communications Letters*, vol. 9, no. 4, pp. 363–365, 2005.
- [16] K. Chen, H. Y. Liu, and X. S. Chen, "Detecting LDoS attacks based on abnormal network traffic," *KSII Transactions on Internet and Information Systems*, vol. 6, no. 7, pp. 1831–1853, 2012.
- [17] K. Chen, H. Liu, and X. Chen, "EBDT: a method for detecting LDoS attack," in *Proceedings of the IEEE International Conference on Information and Automation (ICIA '12)*, pp. 911–916, Shenyang, China, June 2012.
- [18] D. Tang, K. Chen, X. Chen, H. Y. Liu, and X. Li, "Adaptive EWMA Method based on abnormal network traffic for LDoS attacks," *Mathematical Problems in Engineering*, vol. 2014, Article ID 496376, 11 pages, 2014.
- [19] M. Yu, "An adaptive method for source-end detection of pulsing DoS attacks," *International Journal of Security and its Applications*, vol. 7, no. 5, pp. 279–288, 2013.
- [20] Y. Xiang, K. Li, and W. Zhou, "Low-rate DDoS attacks detection and traceback by using new information metrics," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 426–437, 2011.
- [21] Z. Zeng, J. Tu, B. Pianfetti et al., "Audio-visual affect recognition through Multi-stream Fused HMM for HCI," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 967–972, San Diego, Calif, USA, June 2005.
- [22] H. Pan, S. E. Levinson, T. S. Huang, and Z.-P. Liang, "A fused hidden Markov model with application to bimodal speech processing," *IEEE Transactions on Signal Processing*, vol. 52, no. 3, pp. 573–581, 2004.
- [23] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden Markov models for complex action recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 994–999, San Juan, Puerto Rico, June 1997.
- [24] L. K. Saul and M. I. Jordan, "Mixed memory Markov models: decomposing complex stochastic processes as mixtures of simpler ones," *Machine Learning*, vol. 37, no. 1, pp. 75–87, 1999.
- [25] L. R. Rabiner, "Tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [26] D. Zhou, H. Zhang, S. Zhang, and X. Hu, "DDoS attack detection method based on hidden Markov model," *Computer Research and Development*, vol. 42, no. 9, pp. 1594–1599, 2005.
- [27] Z.-J. Wu and D. Zhang, "Attack simulation and signature extraction of low-rate DDoS," *Tongxin Xuebao/Journal on Communications*, vol. 29, no. 1, pp. 71–76, 2008.
- [28] H.-P. Hu, J. Zhang, B. Liu, L. Chen, and X. Chen, "Simulation and analysis of distributed low-rate denial-of-service attacks," in *Proceedings of the 5th International Conference on Computer Sciences and Convergence Information Technology (ICCIT '10)*, pp. 620–626, IEEE, Seoul, Republic of Korea, December 2010.
- [29] J. Zhang, H.-P. Hu, B. Liu, and F.-T. Xiao, "Detecting LDoS attack based on ASPQ," *Journal on Communications*, vol. 33, no. 5, pp. 79–84, 2012.

- [30] P. D. Welch, "The use of the fast Fourier transform for estimation of spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on Audio and Electroacoustics*, vol. 15, no. 2, pp. 70–74, 1967.
- [31] H. C. So, Y. T. Chan, Q. Ma, and P. C. Ching, "Comparison of various periodograms for sinusoid detection and frequency estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 3, pp. 945–952, 1999.
- [32] J. Mirkovic and P. Reiher, "D-WARD: a source-end defense against flooding denial-of-service attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 3, pp. 216–232, 2005.
- [33] S. Kasera, J. Pinheiro, C. Loader, M. Karaul, A. Hari, and T. LaPorta, "Fast and robust signaling overload control," in *Proceedings of the 9th International Conference on Network Protocols (ICNP '01)*, pp. 323–331, Riverside, Calif, USA, November 2001.
- [34] J. Francois, S. Wang, R. State et al., "BotTrack: tracking botnets using NetFlow and PageRank," in *NETWORKING 2011*, vol. 6640 of *Lecture Notes in Computer Science*, pp. 1–14, Springer, Berlin, Germany, 2011.
- [35] H. L. Jiang, X. L. Shao, and Y. F. Li, "Online botnet detection algorithm using MapReduce," *Journal of Electronics and Information Technology*, vol. 35, no. 7, pp. 1732–1738, 2013.
- [36] S. Nagaraja, P. Mittal, C. Hong et al., "BotGrep: finding P2P bots with structured graph analysis," in *Proceedings of the 19th USENIX Conference on Security*, Washington, DC, USA, 2010.

Review Article

Survey of Security Technologies on Wireless Sensor Networks

Qiuwei Yang,¹ Xiaogang Zhu,² Hongjuan Fu,¹ and Xiqiang Che³

¹College of Information Science and Engineering, Hunan University, Changsha, China

²School of Computer Science and Information Engineering, Hubei University, Wuhan, China

³ChangSha LeGou Network Technology Co. Ltd., Changsha, China

Correspondence should be addressed to Xiaogang Zhu; zxcg@hubu.edu.cn

Received 14 October 2014; Accepted 14 December 2014

Academic Editor: Chin-Chen Chang

Copyright © 2015 Qiuwei Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Because of their low cost and adaptability, wireless sensor networks are widely used in civil, military, and commercial fields and other fields. However, since the sensor node in the calculation of the capacity, battery capacity, and storage capacity are restricted by the limitations and inherent characteristics of the sensor networks, compared to traditional networks, which makes wireless sensor networks face more security threats. This paper summarized research progress of sensor network security issues as three aspects, key management, authentication, and secure routing, analyzed and commented on these results advantages and disadvantages and pointed out the future direction of the hot research field.

1. Introduction

The rapid developments in wireless communication, sensor technology, and embedded computing technology have promoted the emergence and development of wireless sensor networks (WSN). Wireless sensor networks consist of a large number of cheap micro sensor nodes deployed in the monitoring area, which is a multihop self-organizing network system formed by wireless communication method, whose purpose is to sense, collect, and process cooperatively the information sensed by sensors in the network distributed area and then forward the results to its users.

Wireless sensor networks, as an emerging network technologies, have risen gradually recently. They can obtain a lot of detailed and reliable information in the network distributed area anytime and anywhere; thus, they are widely used in military defense, industry, agriculture, construction and urban management, biomedical and environmental monitoring, disaster relief, public safety and antiterrorism, hazardous and harmful regional remote control, and so on which are much accounted by many governments. Wireless sensor networks have a very important scientific and practical value.

However, the wireless sensor networks are usually deployed in harsh environments, such as no region or enemy positions in addition the energy, bandwidth, data processing, storage capacity, and other factors of wireless network are limited, which make wireless sensor networks vulnerable to attack. The security of wireless sensor networks is of great social concern. In particular in some important areas (such as military target detection and tracking), once the sensor network is attacked or destroyed, this would likely lead to disastrous consequences. Therefore, the way to design security mechanisms can provide confidentiality protection and authentication features to prevent malicious attacks and create a relatively safe working environment for sensor networks, which is a key issue of whether the wireless sensor networks are practical. Therefore, the issues and challenges faced by wireless sensor network security technology are becoming the main research area all over the world.

In recent years, a lot of research work on key aspects of wireless sensor network security key, protocols, algorithms, architecture, and so on has been done and has made many achievements. This paper summarized research progress situation of sensor network security issues from key management, authentication, and secure routing, three aspects, by analyzing and commenting on these results advantages

and disadvantages and pointed out the direction of future research to explore new solutions.

2. The Basis of Wireless Sensor Networks Security Theory

2.1. The Characteristics of Wireless Sensor Networks. Wireless sensor networks, as a special ad hoc network, compared with other wireless networks, mainly have the following characteristics [1, 2].

- (i) *No Central Node.* Wireless sensor network has no absolutely central node, and all nodes are in equal status. Not only is it the gatherers of information but also the forwarders for other nodes transfer information. Network nodes coordinate behavior with each other through a distributed algorithm.
- (ii) *Self-Organization.* A wireless sensor network requires every sensor node to be independent and flexible enough to be self-organizing and self-healing according to different situations. There is no fixed infrastructure available for the purpose of network management in a sensor network. This inherent feature brings a great challenge to wireless sensor network security as well. If self-organization is lacking in a sensor network, the damage resulting from an attack or even the hazardous environment may be devastating.
- (iii) *Large-Scale.* A wireless sensor network usually consists of thousands of tiny sensors, not primarily depending on the ability to upgrade individual devices but to improve the reliability and stability of the system depending on large-scale and redundancy of embedded devices to work together.
- (iv) *Volatility of Network Topology.* In wireless ad hoc network, various factors such as node mobility and decrease of the remaining power of the power control box in the sensor nodes can lead to network topology changes and make the network topology constantly change which is not regular and unpredictable.
- (v) *Multihop Routing.* Wireless sensor networks use multihop routing mechanism. Due to the limits of transmitting power and the communication coverage radius, when communicating with other nodes out of the coverage, the node needs the intermediate nodes to forward.
- (vi) *Data-Centric Networks.* As nodes are randomly deployed, the relationship between the network and node number is entirely dynamic, showing up that there is no necessary connection between the node number and the node position. User directly reports the events of interest to the networks; then the networks report the information accessed in a specified time to the user. Therefore, the wireless sensor network is a data-centric network.

2.2. WSN Security Needs. According to their own characteristics, the wireless sensor networks differ from the traditional wireless networks, facing more demands especially in terms of security. In order to resist different kinds of security attacks and threats and to ensure the confidentiality of the tasks performed, the reliability of data generated, the correctness of data fusion, and the security of data transmission, the security requirements are mainly in the following areas [3].

- (1) *Data Confidentiality.* Data confidentiality is an important network security need requiring that all sensitive information in the storage and transmission process must ensure its confidentiality. Divulging the content of the information to any unauthorized user is not allowed.
- (2) *Data Integrity.* With the assurance of confidentiality, an attacker could not get the real content of information, but the recipient does not guarantee that the data it receives is correct, because malicious intermediate nodes can intercept, tamper, or disturb the information during the transmission. Through data integrity identification, you can ensure that the data won't change anymore during its transference process.
- (3) *Data Freshness.* Data freshness view is to emphasize that each of the received data is the latest from the sender, which makes it stop receiving repeated information. The main purpose to ensure the freshness of the data is to prevent replay attacks.
- (4) *Availability.* Availability requires the sensor networks that can always provide information access service to the legitimate users according to the preset. But the attacker can make some or all of the sensor network paralyzed by forging and interfering signal or other methods to destroy availability of the system, such as DoS (denial of service) attacks.
- (5) *Robustness.* Wireless sensor networks are highly dynamic and uncertain, including changes in the network topology and the nodes' disappearing or joining. Therefore, the wireless sensor networks under a variety of security attacks should have strong adaptability, and even if a particular attack succeeds, the performance can make the impact minimized.
- (6) *Access Control.* Access control requires the ability to identify the users who access wireless sensor networks to ensure the legitimacy. Access control determines who can access the system, what system resources can be accessed, and how to use these resources.

3. The Research Progress of Wireless Sensor Networks Security

3.1. Key Management. Due to the characteristics of the wireless sensor networks, many mature key management schemes in traditional wireless networks cannot be directly applied to wireless sensor networks. In the security solutions for wireless sensor networks, encryption technology is the basis for a number of security technologies, by encrypting wireless sensor networks that can meet the needs of certification,

confidentiality, nonrepudiation, integrity, and so on. For encryption technology, key management is the key issue to be resolved.

3.1.1. Key Management Schemes Classification. In recent years, researchers have proposed many key management schemes. There can be a variety of categories for these schemes according to the characteristics of them. According to the cryptosystem they used, they can be divided into symmetric and asymmetric key management schemes. According to key distribution methods of the node, they can be split into random key management schemes and deterministic key management schemes. According to the network topology, they can be divided into distributed key management schemes and hierarchical key management scheme, and so on [4].

(1) Symmetric and Asymmetric Key Management. Depending on the difference of cryptosystem, the wireless sensor network key management can be divided into symmetric key management and asymmetric key management. In symmetric key management, the encryption and decryption key of the sensor node are the same, which is simple, and it has a small calculation and storage amount. Comparing with the asymmetric key, the symmetric key has an advantage in terms of computational complexity, but it is inadequate in the aspects of key management and security. Asymmetric key management has been considered unsuitable for wireless sensor networks, mainly due to its relatively high requirement for computing, storage, and communication capabilities of nodes. But with the gradual deepening of the relevant studies, some asymmetric encryption algorithms can now be applied in wireless sensor networks.

(2) Random and Deterministic Key Management. According to the difference of the method in which nodes obtain the key, the key management in wireless sensor network can be split into random key management and deterministic key management. In the random key management, sensor nodes get their keys from the key pool or multiple keys space by random sampling. In deterministic key management, sensor nodes calculate the determination probability to get their keys. The advantages of random key management are a relatively simple way to get the key and the flexible deployment, and its disadvantage is that there may exist part of useless key information in the sensor nodes. The advantages of deterministic key management are that it can obtain more accurate key and the session key can be established directly between any two sensor nodes. Its disadvantage is that flexibility of deployment decreases and computational overhead of key negotiation becomes large.

(3) Distributed and Hierarchical Key Management. Depending on the topology of network, the wireless sensor network key management can be divided into distributed key management and hierarchical key management. In distributed key management, the computation and communication capabilities of sensor nodes are the same, and the key negotiation and update are completed through the mutual

cooperation between sensor nodes. In hierarchical key management, network nodes are split into clusters, and each cluster is composed of cluster head and ordinary sensor nodes. The ordinary sensor nodes complete key distribution, consultation, and update through their cluster head. The characteristic of distributed key management is that the neighboring nodes collaborate to achieve key negotiation. The feature of hierarchical key management is that the requirement of computation and storage capacities of the common nodes is not too high, but once the cluster head is captured by the attacker, it will threaten the security of the entire network.

3.1.2. The Typical Schemes of Key Management. Eschenauer and Gligor [5] first proposed a key management scheme for distributed sensor networks. The basic idea of the program is that a large key pool with the total number of the key is S and key identifier are generated first, each node could select m ($m \ll S$) different keys from the key pools randomly; such randomly preassigned manner made any two nodes have a certain probability of existing shared key. If there are shared keys between two adjacent nodes, then select one randomly as the paired key of the two sides to establish a secure channel. Otherwise, the entered node establishes a key path of the two sides through other neighboring nodes which exist shared key after several jumps. The advantages of E-G scheme are mainly reducing the key storage pressure of each node and suitable for large-scale WSN key management. But there are also disadvantages of this program, and its security communication is uncertainly because the establishment of shared key is based on probability.

On the basis of E-G scheme, Chan et al. [6] proposed a q -composite random key predistribution scheme. The specific implementation process of q -composite random key predistribution scheme is basically similar to E-G scheme, except that the E-G program just selects a public key as the main shared key between two nodes, while the q -composite scheme requires that two adjacent nodes can establish the main shared key after the deployment only when there is at least q shared key between them. Compared with the E-G scheme, the q -composite program improves the capacity of resisting capture attacks of nodes but increases the overlap degree of shared key between the nodes and limits the scalability of the network.

Zhu et al. [7] thought that any single key mechanism could not achieve the security needs of wireless sensor networks, so they proposed a LEAP protocol based on multiple key mechanisms to establish secure communications. The protocol maintained four keys in each node: a globally key shared with the base station, a group key shared with all nodes within the network, a paired key shared with neighboring nodes, and a cluster head key shared with the cluster head. Compared with the random key predistribution protocol, the nodes' computation loads and storage space requirements for the LEAP protocol will increase, but it can guarantee that there is a shared key between the nodes needed to exchange data and support a variety of network communication modes.

Donggang et al. [8] proposed a key distribution scheme based node group. The basic idea of the program is that

assuming that the system previously generates a large key pool S , then the S is into several subkey pools, making sure that each node deployment group has a corresponding subkey pool, such as S_1, S_2, S_3, \dots , and that the size of each subkey pool was $|S_1|$. Then the nodes of each deployment group select some keys from the corresponding subkey pool randomly. Due to the fact that the establishment of the secure channel between nodes needs at least one shared key, so it requires there should be public key between the corresponding subkey pools of neighborhood groups to ensure the connectivity between nodes. Donggang et al. set up the repetition factor of the same key between the corresponding subkey pools of adjacent groups. The scheme is more safe, after the node is under attack, it has little impact on the security of other nodes in the network, but the storage overhead of such key management scheme is large; for the resource-constrained wireless sensor networks it is a very serious problem.

Du and Guizani et al. [9] think that many key management schemes which based on symmetric key considered too much about network connectivity and hope to find a method that any two nodes can get shared key while ignore the communication of the two nodes. In the context of heterogeneous sensor networks, they proposed a route-drive public key management scheme based lightweight ECC, which only allocated communication key for neighbor node. The performance simulation shows that, compared to the symmetric key mechanism, this scheme significantly improved safety and also saved energy and storage space compared to key management schemes of other asymmetric key mechanisms.

3.2. Certification. Network security certification is another important part of the network. It includes identity authentication and message authentication, and methods used are symmetric encryption and asymmetric encryption method. This section summarizes research work on the two modes of certification.

3.2.1. Identity Authentication. Wireless sensor nodes are deployed to work after the domain, on the one hand, to ensure that users have the legal status to join the network, and, on the other hand to effectively prevent unauthorized users from joining, so the wireless sensor network authentication mechanism must be used to determine the user's identity legitimacy. By using legitimate authentication of neighboring nodes or nodes and base stations. Wireless sensor network provides secure access mechanism, when all nodes access the self-organizing network. There are currently certified questions symmetric encryption algorithm based authentication methods and authentication methods based on asymmetric encryption algorithms.

(1) Authentication Based Symmetric Encryption Algorithm. In wireless sensor networks, due to the limited energy of nodes, the nodes of computing power and communication bandwidth, computational overhead of symmetric cryptosystem is much smaller than the asymmetric cryptosystem. Considered from the perspective of resource conservation, the symmetric

cryptosystem is the most suitable characteristics for wireless sensor networks.

In 2002, Eschenauer and Gligor [5] first proposed a configuration scheme of shared key that is a symmetric key management, and then secure communication can be established through the preshared key between any pairs of nodes, and also can authenticate the identity of each other. Eschenauer and Gligor's scheme has lowly computational complexity and storage burden but lack of security of the scheme.

In 2003, Chan et al. [6] proposed q -composite random key predistribution scheme based on improved E-G scheme. The model requires that the number of public keys is to be up to q ; the program reduces the probability of a certain degree, the session key in wireless sensor networks overlap, and improves the anticapture capabilities and enhanced security. However, to make the part of key overlap reach q and in order to make the probability of the key overlap shared between the neighboring nodes reach preset requirements, thus we require reducing the size of the key pool, so the security may be lowered, and the key is to find a suitable size pool.

In 2005, Bauer and Lee et al. [10] proposed a distributed authentication protocol, using the concept of secret sharing and cryptography groups agree. A network concludes plurality of subgroups; each subgroup equips a base station and subgroups communicate with each other by base station. The program advantage is not employing any high consumption of encryption/decryption program in the certification process, but using the way of secret sharing and group agreed, and its fault tolerance is good; computationally efficient and authentication strength is high. The disadvantage of the program is that all nodes within the subgroup should communicate cooperatively when authenticated, it is likely to cause an information collision when the nodes delivery the determined packet.

In recent years, wireless sensor network authentication in the exploration for symmetric ciphers still did not stop. In 2010 Qiu et al. [11] proposed an efficient extensible authentication protocol to ensure that there is at least one probability of a shared key between two nodes and update the authentication key based on dynamically changing wireless sensor networks. The program has low storage overhead and energy consumption and does not cause much communication overhead. It is suitable for resource-constrained wireless sensor networks.

(2) Authentication Based Asymmetric Encryption Algorithm. Although symmetric cryptosystem has an advantage in the calculation of authentication, it has no strong asymmetric cryptography in terms of safety, and after the elliptic curve cryptosystem proposed, many studies show that even if there is a defect that the amount of computation and storage load are too large, asymmetric keys are still available for wireless sensor networks, asymmetric keys can still be used for wireless sensor networks. Here are some typical asymmetric cryptography schemes.

Watro et al. [12] proposed TinyPK entity authentication scheme based on the RSA algorithm. TinyPK authentication protocol uses a challenge-response mechanism to be able to

perform authentication for external organizations and safely transmit session key to third parties from wireless sensor network. The program uses a low index RSA algorithm, which to some extent reduces the amount of calculation and storage overhead. Meanwhile, in order to make the program adapt to the limited resources of sensor devices, TinyPK designed a protocol to make general nodes only need to perform fast, small resource consumption data encryption and signature verification work and the energy consumption of a large decryption and signature conducted by the work station with relatively ample energy or an external organization. However, TinyPK scheme security is not high. If a node is captured by the authentication, then the entire network will become unsafe, and when the key length is too long, the computational overhead is great.

Benenson et al. [13] proposed strong user authentication protocol which has improved TinyPK scheme: instead of using the key length which is shorter, use elliptic curve cryptography (ECC) which has shorter key length and with the same security strength; instead of traditional single certification, use n certification. The traditional single-user certification is that as long as one user is certified on Renyiyitai host or node; then the user can obtain legal status to enter the entire network, but n certification requires that users at least pass $n-t$ nodes of its communication range, and then the user can obtain legal status, which improves the security to some extent, but its communication overhead is large and cannot prevent denial of service attacks.

Malone-Lee [14] in 2002 first proposed identity-based signature algorithm (IBS). The IBS is an identity-based signcryption scheme by using a concept of signcryption and based on password system on the basis of identity. However, message in dense text sign of this scheme is visible, which makes the message confidentiality threatened; on this basis, Liqun and John proposed an improved identity-based encryption algorithm [15]. The algorithm includes the creation, extraction, encryption, signature, authentication, and decryption of six stages. The similarity with the general IBE encryption algorithm is that the algorithm signs the message with the sender's private key first and then encrypted with the recipient's public key and sends a message signed to the recipient. After receiving cipher text, receiver decrypts the message first, and then according to the message decrypted to verify whether the message sent by the sender to complete the authentication statement. Simulation results show that Malone-Lee algorithm is greatly improved compared to some other identity-based signed algorithms at that time in terms of safety and performance.

In 2006, Piotrowski et al. [16] proved by experiments that, under certain conditions, a large number of public key authentication schemes which can be applied to wireless sensor networks have been proposed. In 2008, An and Peng [17] proposed Tiny ECC certification program and noted that the program is suitable for wireless sensor network applications. Li et al. [18] proposed a combination of public key-based two-way authentication protocol CPK. In 2009, Das proposed a two-factor authentication scheme [19] and used passwords and smart card way to achieve certification. In 2010, Li-Ping Zhang and Yi Wang proposed an id-based authenticated key

agreement scheme [20] without a certificate. In 2011, Yeh et al. [21] proposed based on ECC encryption secure user authentication protocols. In 2012, Peng [22] proposed a multi-identity-based authentication scheme and Hong et al. [23] proposed a lightweight interactive authentication scheme. In 2013, Shi and Gong [24] overcome the deficiencies of Manik Lal [19] public key authentication scheme and proposed a new encryption based ECC authentication protocol.

3.2.2. Message Authentication. Message authentication means to confirm the message received from sender statement. Message authentication can be achieved by symmetric encryption and digital signature technology. At present, there are mainly two types of message authentication; one is point-to-point message authentication and the other is broadcast authentication. In a point-to-point message authentication, we can use most of ID authentication methods to achieve. In wireless sensor networks, in order to save resources, broadcast is a common method of transmission. Currently, TESLA protocol is the most classic broadcast authentication protocol, and a lot of research work is commenced on the TESLA protocol.

Perrig et al. [25] proposed a miniature and high efficiency time-based stream authentication protocol TESLA (micro version of TESLA) which tolerate loss package. TESLA's main contribution is the use of symmetric key technology, which achieved asymmetric encryption function, and its main idea is to use key distribution delays and one-way hash functions to achieve the irreversibility broadcast authentication. μ TESLA first broadcast a packet through key authentication and then released key. The irreversibility of the one-way hash function can guarantee that no one can get any information authentication key before the release of the key, so there is no way to forge correct broadcast packets before the broadcast packet is certified.

In order to improve μ TESLA, Donggang and Peng [26] proposed a modified TESLA protocol, multistage TESLA protocol. First the multistage TESLA protocol introduced schedule and broadcast initialization parameters method to replace the ways of using thin TESLA security parameters in the initialization process. Secondly, the protocol uses a multilevel secret key chain model and abandoned TESLA using secret key chains to sustain long life cycle approach TESLA. In addition, the protocol uses redundant capacity transport mechanisms and random selection strategy to complete the secret key chain publishing tasks, in order to improve network packet loss tolerance and capability of the fight against DoS attacks. While there are many good features of multilevel TESLA, the realization of the protocol is of high complexity and it takes up more of the node memory and computing resources.

The μ TESLA protocol is designed for single base station sensor networks. YuLong and Qingqi et al. [27] proposed MM TESLA based on the TESLA (multiple-base station multilevel TESLA) protocol. This paper introduces the idea of threshold cryptography and proposed a broadcast authentication protocol MM TESLA which is suitable for multi station sensor network. MM TESLA and its extensions have high success ratio of authentication, high reliability, and tolerance of high

channel error rates and resist the known levels of DoS, DoM, and false message attack characteristics.

3.3. Security Routing. Routing algorithm is the basis of information transmission and convergence in the wireless sensor networks. As multihop networks, wireless sensor networks have special characteristics, especially in the aspect of security routing and the need for in-depth research. At present, domestic and foreign scholars have proposed a variety of wireless sensor network routing protocols. This section we will describe several typical security routing protocols.

3.3.1. Data-Centric Security Routing Protocol. In view of the fact that wireless sensor networks is a data-centric network, the data-centric routing protocol has been designed for wireless sensor networks. The protocol takes into account the problem of data redundancy and obtains the fused data through collaboration between the nodes, thus improving data transmission efficiency and saves network energy.

Joanna and Wendi et al. proposed SPIN protocol [28] which is a data-centric adaptive routing protocol. In wireless sensor networks, since nodes sensing data have certain similarities, SPIN protocol can effectively reduce the amount of data transmitted and energy consumption in the network through negotiation between nodes. However, SPIN protocol needs to send inspection packet before sending the packet every time, thus causing a large data transmission delay. In addition, SPIN data broadcast mechanism cannot guarantee the reliability of data transmission.

Chalermek and Ramesh [29] designed a directed diffusion routing protocol specifically for wireless sensor networks, which was based on data-centric routing protocol model. The protocol introduces a network “ladder” concept, combining with the local routing protocol for wireless sensor networks communication. Directed diffusion process is divided into query diffusion, data dissemination, and path reinforcement. Since the establishment of directed diffusion routing requires a flood spread and causes big expense of energy and time, the algorithm is suitable for the scenario, which has a large number of queries but a short time.

Rumor routing that overcomes the problem of excessive spending from establishing forwarding path thought flood spread method was proposed by David and Deborah [30]. Rumor routing basic idea is that time zone sensor node generates agency messages, and agency messages spread outward diffusion along a random path, while the query messages from the sink node also spread along a random path in the network. When the transmission path of agency messages and query messages cross together, there is a full path from a sink node to the event area. Compared with the directed diffusion routing, rumor routing effectively reduces the routing established expense. However, because of rumors that the routing path is generated randomly, the data transmission path is not optimal path, and maybe even routing loops exist.

3.3.2. Location-Based Secure Routing Protocol. Most location-based routing protocols assumed that each node in the network knows own location information, and location nodes can exchange information with their neighbors, so that nodes can use location information to make routing choice without the need to save the routing table, and the typical protocols are GPSR and GEAR.

GPSR (greedy perimeter stateless routing) routing algorithm is the method by directly using geographic information to establish the routing path, which was proposed by Brad and Hsiang-Tsung from Harvard University [31]. GPSR algorithm uses a greedy routing strategy, and each node only needs to know the destination node of the packet and the location information of the next hop of the candidate node, which can make the right choices to send packet without the need for other network topology information, greatly reducing the consumption of maintaining network information; moreover, it has better fault tolerance and scalability, but the agreement does not take into account energy efficiency, easily lead to excessive use of certain nodes and shorten the life cycle of the network.

Yu et al. from the UCLA University, USA [32], proposed GEAR (geographic and energy aware routing) routing algorithm, which combines the directional diffusion routing and GPSR routing methods, considers the node energy in the route, and thus solves the problem of unbalanced energy consumption in GEAR. GEAR assumes that the position information of the event area is known, and the nodes know their location information and residual energy. In addition, the node via a simple Hello message exchange mechanism will be able to know the location information and residual energy information for all nodes. Routing mechanism based on these address locations and energy information establish the optimal path from aggregation node to the event area to avoid flooding, reducing the overhead of route established. However, due to the lack of sufficient topology information, GEAR may reduce the routing efficiency when encounter routing void in the routing process.

3.3.3. The Security Routing Protocol Based on Hierarchical Structure. Sensor nodes are divided into multiple clusters in the hierarchical routing protocol; each cluster has a cluster head node that can not only control communication between nodes within a cluster, but also gather and fuse data of the cluster area. Then each cluster head node will send the fused data to the gateway node, which can reduce the traffic and maintain node power consumption. Typical routing protocols are LEACH and TEEN.

A research group of Professor Wendi Rabiner et al. from Massachusetts Institute of Technology proposed that LEACH (low energy adaptive clustering hierarchy) protocol [33] is a classic clustering class routing protocol. Each round LEACH algorithm consists of establishment phase and data transmission phase of a cluster head. The algorithm allows nodes in the network balanced energy consumption and prolongs the network life cycle. But LEACH does not guarantee the position and amount of cluster heads in system, which makes the elected cluster heads distributed unevenly.

TEEN [34] was proposed on the basis of LEACH. The basic idea is that a cluster head is selected randomly periodically and equiprobably, and the other noncluster head nodes based on the nearest principle joined in appropriate cluster to form the virtual cluster and make energy of the whole network load evenly distributed to each sensor node, which can reduce network energy consumption and extend the network life cycle. In the process of establishing the cluster, the cluster head node broadcasts hard threshold and soft threshold to the other nodes, which can strike a reasonable balance between accuracy and data transmission network energy consumption by adjusting the two thresholds. Each round TEEN protocol consists of establishment phase and stable data transfer phase of clusters. TEEN agreement by a reasonable set of hard and soft thresholds only transmits the information of interest to users, which can effectively reduce traffic and the power consumption of the system. Simulation studies show that TEEN protocol is more effective than LEACH protocol. But like LEACH protocol, TEEN also will encounter similar Hello flood attacks, selective forwarding attacks, witch attacks, and so on.

3.3.4. The Security Routing Protocol Based on Multipath Transmission. Multipath routing can effectively improve the success rate of data messages submission and balance node energy consumption to prolong the survival time of the node, while multipath routing is an effective prevention method against selective forwarding attacks.

Quadjaout et al. [35] proposed a new multipath routing SMRP and, on this basis, designed the SELF. In SELF, the control nodes in wireless sensor network send a key update command every one given slot. When the normal nodes receive the key update command, they will update their keys and report update result to the control nodes in their own cluster. The control nodes regard the normal nodes which have not updated their own keys in time as captive nodes and send making-invalid broadcast in the cluster. As a result, SELF can prevent the enemy from pretending to be a legal node by making use of the keys of captured sensor nodes.

Aiming at the problem of the traditional anonymous routing protocols being single path, Zhang et al. [36] proposed a multipath protocol MPRASRP and it can effectively prevent attackers from obtaining the identity of the source node and the destination node, thereby preventing attackers from further tracking the information processing among two nodes. The method to guarantee node anonymity is that the identity of the source node and the destination node are encrypted by the destination node public key, and only the destination node can decrypt the packet. The protocol can effectively prevent the middleman attack and even under harsh environmental conditions is also very effective, but the protocol does not prevent replay attacks.

The basic idea of the MSR [37] protocol is that, firstly, the original pieces of information are divided into subdata packets by removing code; then the subdata packets are sent out via the multiple paths. Finally, these pieces of information are combined by destination node. The agreement includes

a random multipath enhanced, passive confirmation and cancellation code. Only when you need to build a random path, passive confirmation can analyze safety behavior of neighbors based on the monitor passive traffic, reduces the routing header, has a good defense against common attacks, and thus guarantees the safety of the route.

4. Some Prospects for Future Research

In recent years, we made a lot of new achievements in wireless sensor network security. However, there are still many shortcomings in security and network applicability. Based on the analysis and summary of WSN security research work above, this section will make a few prospects for future research from key management, authentication, and secure routing, three aspects.

(1) Key Management

- (i) *More Efficient Asymmetric Key System.* As the features of wireless sensor network node resource are limited, asymmetric key system has been considered unsuitable for wireless sensor networks; after elliptic curve cryptography (ECC) has been proposed, asymmetric key system application in wireless sensor networks has become possible. Compared to the symmetric key system, asymmetric key system has great advantages in terms of management and security keys. In recent years, many scholars have proposed many effective key management schemes based on ECC public key infrastructure. The space of research in this area is still large in future, and the application prospects are very broad.
- (ii) *Apply to Key Management Scheme of Network.* In recent years, the development of network technology puts forward higher requirements and challenges for wireless sensor network security such as the perception of information privacy issues, problems with the mobile network and Internet security systems integration, and controlling reliability issues. Obviously, the key management system which is suitable for the Internet of Things must be able to meet the security needs above. This also provides researchers with a more research space.

(2) Certification

- (i) *The Establishment of a Trust Model.* Establishing trust model between nodes can reduce communication overhead between nodes and improve the efficiency of certification to some extent. It can greatly reduce system overhead to improve the network lifetime, particularly through large-scale network of trust mechanism.
- (ii) *Improved Public Key Algorithm.* The public key algorithm has its advantages in terms of safety, but the calculation is too big and there is some difficulty in resource-constrained wireless sensor networks applications. How to reduce the computational complexity

of this section is one of the hot and difficult current researches.

(3) Security Routing

- (i) *Multipath Routing Protocol*. In recent years, traditional single-layer sensor network only concerns the effective use of energy node, but it does not consider security issues. The researchers began to focus on multipath routing protocol security and research how to avoid the secure routing protocols which has been the victim node and the potential victim node, but currently the secure routing protocols consider the situation in which the victim node has been detected. For the victim nodes have not yet been detected, secure routing protocols can be a direction on how to avoid future research.
- (ii) *Routing Protocol Hierarchy*. In addition, due to the particularity of the hierarchical sensor networks structure, they are convenient for security solutions while providing additional management overhead and other issues hierarchy; therefore, routing protocol hierarchy is a research focus in recent years.

Conflict of Interests

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

Acknowledgments

This work was sponsored by the National Natural Science Foundation of China (no. 61300036), Projects in the National Science & Technology Pillar Program (no. 2013BAH38F01), and the Foundation for University Key Teacher by the Ministry of Education, China.

References

- [1] J. P. Walters, Z. Q. Lian, W. S. Shi et al., "Wireless sensor network security: a survey," in *Security in Distributed, Grid, Mobile, and Pervasive Computing*, Auerbach Publications, Boca Raton, Fla, USA, 2006.
- [2] M. Sharifnejad, M. Sharifi, M. Ghiasabadi, and S. Beheshti, "A survey on wireless sensor networks security," in *Processing of the 4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications*, Hammamet, Tunisia, March 2007.
- [3] P. G. Shah, "Network security protocols for wireless sensor networks: a survey," <http://www.niitcrs.com/iccs/papers/2005.42.pdf>.
- [4] S. Qian X, *The Key Management Scheme in Wireless Sensor Networks*, University of Science and Technology of China, Hefei, China, 2009.
- [5] 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.
- [6] 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, Washington, DC, USA, May 2003.
- [7] 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, ACM Press, New York, NY, USA, October 2003.
- [8] D. Liu, P. Ning, and W. Du, "Group-based key pre-distribution in wireless sensor networks," in *Proceedings of the ACM Workshop on Wireless Security*, pp. 11–20, 2005.
- [9] X. J. Du, M. Guizani, Y. Xiao, and H.-H. Chen, "Transactions papers a routing-driven Elliptic Curve Cryptography based key management scheme for Heterogeneous Sensor Networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1223–1229, 2009.
- [10] K. Bauer and H. Lee, "A Distributed Authentication Scheme for a Wireless Sensing System," in *Proceedings of the 2nd International Workshop on Networked Sensing System*, pp. 210–215, San Diego, Calif, USA, 2005.
- [11] Y. Qiu, J. Y. Zhou, J. Baek et al., "Authentication and key establishment in dynamic wireless sensor networks," *Sensors*, vol. 10, no. 4, pp. 3718–3731, 2010.
- [12] R. Watro, D. Kong, S.-F. Cuti, C. Gardiner, C. Lynn, and P. Kruus, "TinyPK: securing sensor networks with public key technology," in *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 59–64, Washington, DC, USA, October 2004.
- [13] Z. Benenson, N. Gedicke, and O. Raivio, "Realizing robust user authentication in sensor networks," in *Proceedings of the Workshop on Real-World Wireless Sensor Networks*, pp. 135–142, Stockholm, Sweden, June 2005.
- [14] J. Malone-Lee, Identity-based signcryption, cryptology ePrint archive, <http://eprint.iacr.org>.
- [15] L. Chen and J. Malone-Lee, "Improved identity-based signcryption," in *Public Key Cryptography—PKC 2005: 8th International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23–26, 2005. Proceedings*, vol. 3386 of *Lecture Notes in Computer Science*, pp. 363–379, Springer, Berlin, Germany, 2005.
- [16] K. Piotrowski, P. Langendoerfer, and S. Peter, "How public key cryptography influences wireless sensor node life-time," in *Proceedings of the 4th ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 169–176, 2006.
- [17] 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, Saint Louis, Mo, USA, April 2008.
- [18] J. J. Li, L. Tan, and D. Y. Long, "A new key management and authentication method for WSN based on CPK," in *Proceedings of the International Colloquium on Computing, Communication, Control, and Management, (CCCM '08)*, vol. 2, pp. 486–490, Guangzhou, China, August 2008.
- [19] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1086–1090, 2009.
- [20] 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.

- [21] 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.
- [22] S. W. Peng, "An ID based multiple authentication schemes against attacks in wireless sensor networks," in *Proceedings of IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS '12)*, pp. 1436–1439, 2012.
- [23] Y. H. Hong, Y. H. Zeng, and Y. J. Huang, "Mutual message authentication protocol in wireless sensor networks," in *Proceedings of the International Conference on Intelligent Information and Networks*, pp. 121–126, 2012.
- [24] W. Shi and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 730831, 7 pages, 2013.
- [25] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks Journal*, vol. 8, no. 5, pp. 521–534, 2002.
- [26] D. Liu and P. Ning, "Multi-level μ TESLA: a broadcast authentication system for distributed sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 3, no. 4, pp. 800–836, 2004.
- [27] Y. L. Shen, Q.-Q. Pei, and J.-F. Ma, "MM μ TESLA: broadcast authentication protocol for multiple-base-station sensor networks," *Chinese Journal of Computers*, vol. 30, no. 4, pp. 539–546, 2007.
- [28] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceeding of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, pp. 174–185, Seattle, DC, USA, August 1999.
- [29] 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*, pp. 56–57, Boston, Mass, USA, 2000.
- [30] D. Braginsky and D. Estrin, "Rumor routing algorithm for sensor networks," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, pp. 22–31, Atlanta, Ga, USA, 2002.
- [31] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pp. 243–254, Boston, Mass, USA, August 2000.
- [32] Y. Yu, R. Govindan, and D. Estrin, "Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks," Tech. Rep. UCLACSD TR-01-0023, 2001.
- [33] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless micro sensor networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Maui, Hawaii, USA, 2000.
- [34] A. Manjeshwar and D. P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," in *Proceedings of 15th International Parallel and Distributed Processing Symposium Workshops (IPDPS '01)*, Issues in Wireless Networks and Mobile, pp. 2009–2015, San Francisco, Calif, USA, April 2000.
- [35] A. Ouadjaout, Y. Challal, N. Lasla, and M. Bagaa, "SEIF: secure and efficient intrusion-fault tolerant routing protocol for wireless sensor networks," in *Proceedings of the 3rd International Conference on Availability, Security, and Reliability (ARES '08)*, pp. 503–505, IEEE Computer Society, Barcelona, Spain, March 2008.
- [36] Z. M. Zhang, C. G. Jiang, and J. Deng, "Multiple-path redundancy secret anonymous routing protocol for wireless sensor networks," in *Proceedings of the 6th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '10)*, pp. 1–4, IEEE, Chengdu, China, September 2010.
- [37] M. A. Moustafa, M. Youssf, and N. M. El-Dering, "MSR: a multipath secure reliable routing protocol for WSNs," in *Proceedings of the 9th IEEE/ACS International Conference on Computer Systems and Applications (AICCSA '11)*, pp. 54–59, Sharm El-Sheikh, Egypt, December 2011.

Research Article

Research on Handoff Delay and Mobility Management Cost of Mobility Protocols in Wireless Sensor Networks

A. Q. Zhao and Y. Hu

School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to A. Q. Zhao; aqzhao@bjtu.edu.cn

Received 25 November 2014; Revised 18 February 2015; Accepted 4 March 2015

Academic Editor: Fei Yu

Copyright © 2015 A. Q. Zhao and Y. Hu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An appropriate network model and some suitable performance evaluation criterions including handoff delay and mobility management cost were proposed in this paper. And in this base the performance of Mobile IP protocol and various micromobility protocols was comprehensively compared and investigated. The research results show that the performance is mainly influenced by two factors which are route update methods of mobility support protocols and mobile network parameters. The route update time and mobility management cost of micromobility protocol are obviously shorter than that of Mobile IP. In all researched micromobility protocols, the route update method of Mobile IP Regional Registration protocol has the optimal performance.

1. Introduction

Mobile IP protocol [1–3] is the most basic mobile support protocol of the Internet. With the development of network and application, Mobile IP is increasingly exposed to serious performance defects. Therefore, micromobility protocols were proposed, such as micromobility protocols, Mobile IP Regional Registration (MIP-RR) protocol [4, 5], Cellular IP protocol (CIP) [6, 7], and HAWAII protocol [8, 9].

Handoff is the most basic operation of the mobile network and the most important content of mobility support protocol research. The handoff performance has a crucial impact on the performance of the mobile network [10–13]. Network layer handoff delay is the most important indicator to measure mobility support protocols, and other handoff performance indicators, for example, packet loss and application throughput decline when making handoff, are all relate to handoff delay. This paper focused on performance evaluation of Mobile IP and a variety of micromobility protocol network layer handoff delay.

To ensure mobility support protocols' use in practice, we believe that the other performance indicator, that is, mobility management cost, is as important as handoff performance. Mobility management cost of mobility support was studied in [14–17]. Paper [14] studied theoretically the influence that the

packets arrival intervals have on the mobility management cost of micromobility protocols and pointed out that the factors affecting its performance is the average packet arrival intervals, rather than the type of interval time distribution. Paper [15] compared the management cost of Mobile IP and micromobility protocols mainly through simulation method.

In the mobility support protocols handoff performance study, we put forward an idea. Firstly a suitable network model was proposed, analyze the factors affecting handoff performance on the basis of the network model, and theoretically compare various mobility support protocols' handoff performance. This paper focused on the research of mobility management cost of mobility support protocols and made analysis and comparison of mobility management cost among various mobility support protocols.

2. Network Modeling

The wireless access network is constituted by several administration domains [13]. And every administration domain is connected to the Internet through GW (Gateway). The cellular model adopted in the administration domain is shown in Figure 1. It is a concentric circle model (the usually used hexagonal model is not adopted in this paper for calculating

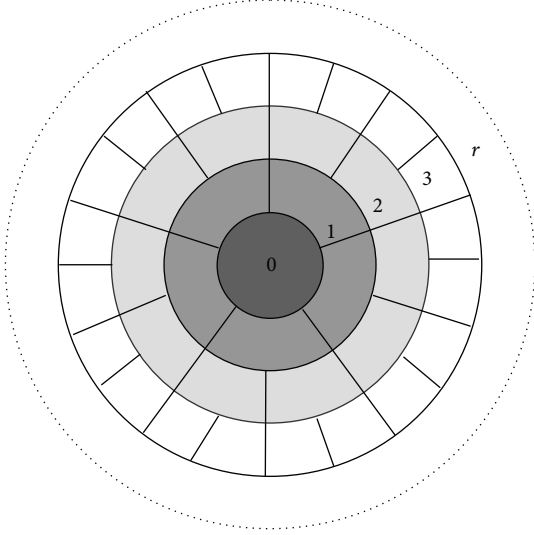


FIGURE 1: Cellular model of the administration domain.

simplification). The model has a center cell (defined as the cell in layer 0). Other cells are distributed around the center cell with their layer number of 1, 2, 3, and so on. There are 5 cells in layer 1. The number of the cell in layer $i + 1$ ($i = 1, 2, 3, \dots$) is two times more than the number of the cell in layer i . Each cell has five adjacent cells. The cell in layer 0 adjoins 5 cells in layer 1, while the cell in layer i ($i = 1, 2, 3, \dots$) adjoins 1 cell in layer $i - 1$, 2 cells in layer i , and 2 cells in layer $i + 1$. A BS (Base Station) which has the function of MRA (Mobile Routing Agent) [18] is set in each cell. We make the BS in layer 0 Gateway of the whole administration domain. All adjacent BSs are linked by wired links. And the distance between them is 1 hop.

Suppose the radius of administration domain is r ; the MHs are uniformly distributed in the cells in administration domains. We denote the probability that the MH occurs in a cell in layer i by p_i . So p_i is given by

$$p_i = \frac{N_i}{N} = \frac{N_i}{\sum_{i=0}^r N_i} = \begin{cases} \frac{1}{(5 * 2^r - 4)} & i = 0 \\ 5 * \frac{2^{i-1}}{(5 * 2^r - 4)} & i = 1, 2, \dots, r. \end{cases} \quad (1)$$

In the formula, N_i and N denote the number of the cell in layer i and the number of the cell in the whole administration domain, respectively.

Suppose the distance from a i -layer cell to GW is i ; then the average distance between MH-cells and GW is given by

$$d = \sum_{i=0}^r i * p_i = \frac{5[(r-1)2^r + 1]}{(5 * 2^r - 4)}. \quad (2)$$

Suppose the probabilities in which a MH executes handoffs to its five adjacent cells are equal. The probability that

the MH executes a handoff from layer i ($i = 1, 2, 3, \dots, r$) to layer j is

$$p_{i,j} = \begin{cases} \frac{1}{5} & j = i - 1 \\ \frac{2}{5} & j = i \\ \frac{2}{5} & j = i + 1. \end{cases} \quad (3)$$

MH uses announcement packets that are broadcasted by BS periodically to trigger network layer handoff; the handoff process is as follows: before the handoff, MH sends and receives data through the old BS. When MH detected the need to handoff, it disconnects the connection with the old BS at first and then establishes a connection with the new BS. MH sends the route update message to update the location information of itself when it receives the first announcement packet from the new BS. After the completion of the route update, MH can send and receive data through the new BS.

3. Handoff Delay

Handoff process shows that the network layer handoff delay consists of two parts: one is the elapsed time from MH disconnecting the connection with the old BS to MH receiving the first announcement packet from the new BS, called movement detection time. The second is the time of MH sending the route update message to update the location information of itself, called route update time. Movement detection time is related to link layer handoff time and the cycle of BS broadcast announcement packets, rather than the mobility support protocols. However, route update time we will focus on hereinafter is determined by route update methods of mobility support protocols.

Route update time consists of two parts. One is the route update packet transmission delay. The other is the route update packet processing delay of mobile support node. Suppose, in the administration domain, the transmission delay of the route update packets per unit distance is T ; the total one-way transmission delay of route update packets in the Internet is $w * T$; processing delay per route update packet of mobile support node is $p * T$. We first compared the route update time of Mobile IP and that of micromobility protocols and then the route update time of different micromobility protocols.

3.1. Comparison of Mobile IP with Micromobility Protocols. In the case of using Mobile IP, route update packets from MH were passed to the new BS and then the GW, arriving at the local agent at last. The reply packets along the opposite path return to MH. We denote route update time by T_{MIP} . T_{MIP} is

$$T_{MIP} = [2(d' + 1) + 2w + 3p]T. \quad (4)$$

In the formula above, d' denotes the average distance between new BS and GW; from formula (2) and (3), we could get d' which is

$$d' = \sum_{i,j} (d + j - i) * p_{i,j} = d + \frac{1}{5}. \quad (5)$$

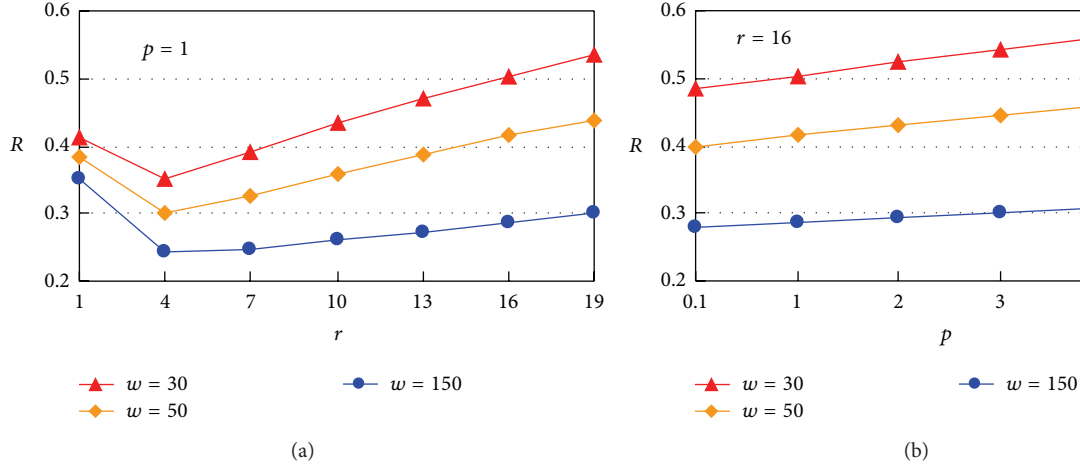


FIGURE 2: Performance of Mobile IP and micromobility protocols.

In the case of using micromobility protocols (with two-layer mobile as an example), route update methods are different between interdomain handoff and intradomain handoff. Interdomain handoff is processed just as that of Mobile IP; route update time T_{inter} is

$$T_{\text{inter}} = [2(r+1) + 2w + 5p] T. \quad (6)$$

Micromobility protocols were used to process intradomain handoff. The route update packets from MH were passed to the new BS and then arrive at GW. The reply packets along the opposite path return to MH. We denote route update time by T_{intra} . T_{intra} is

$$T_{\text{intra}} = [2(d' + 1) + 3p] T. \quad (7)$$

Interdomain handoff occurs only when the MHs are in the administration domain boundary cells. According to formula (1) and (3), the probability of MH interdomain handoff is

$$p_{\text{inter}} = p_r * p_{r,r+1} = \frac{2^r}{(5 * 2^r - 4)}. \quad (8)$$

We got the average route update time of micromobility protocols from formula (6), (7), and (8):

$$T_{\text{MMP}} = p_{\text{inter}} * T_{\text{inter}} + (1 - p_{\text{inter}}) * T_{\text{intra}}. \quad (9)$$

To compare the route update time of Mobile IP protocol and micromobility protocols, we calculated the ratio of the two as follows:

$$R = \frac{T_{\text{MMP}}}{T_{\text{MIP}}}. \quad (10)$$

We take $T = 1$; when w takes different values, ratio R of the route update time was calculated with the change of p and the value of the administration domain radius r . The result is shown in Figure 2.

Figure 2 shows that $R < 1$ in any case; this means that the introduction of micromobility protocols is necessary because the average route update time of micromobility protocol is obviously shorter than that of Mobile IP. When r is small, R is at a middle level; as r increases, R decreases rapidly and soon reaches the minimum; then, as the further increases of r , R gradually increases slowly. The change trend above indicates that the advantage of micromobility protocols is related to the size of the administration domain; there is an optimal r which can be used as one of the considerations to determine administration domain size. Figure 2 also shows that R declines as w increases. For the larger w is, the farther away MH from the local network is, so the farther away MH from the local network, the more obvious advantages of micromobility protocols. The introduction of micromobility protocols result to the nodes that process route update packets increases, so, with p increasing, the advantage of the introduction of micromobility protocols decreases.

3.2. Comparison of Different Micromobility Protocols. We compared the route update time of MIP-RR, CIP, and HAWAII protocol only in situation of intradomain handoff, for micromobility protocols all adopt Mobile IP protocol to process interdomain handoff. Suppose MH is in an i th layer cell of the administration domain before handoff ($i = 1, 2, \dots, r - 1$); the packet forwarding path of MH within domain is the optimal path from GW to the current BS.

In the case of MIP-RR protocol, route update packets from MH were sent to GW via new BS when handoff happens, ending at cross MRA (the lowest level of public MRA on both new route and old route of MH) [6]. The reply packets along the opposite path return to MH. If MH handoff is from the i th layer to the $(i-1)$ th layer or the $(i+1)$ th layer, cross MRA will be the new BS or the old BS, respectively. Their route update times are calculated as follows:

$$\begin{aligned} T_{\text{MIP-RR}(i,i-1)} &= (2 + p) T, \\ T_{\text{MIP-RR}(i,i+1)} &= (4 + 3p) T. \end{aligned} \quad (11)$$

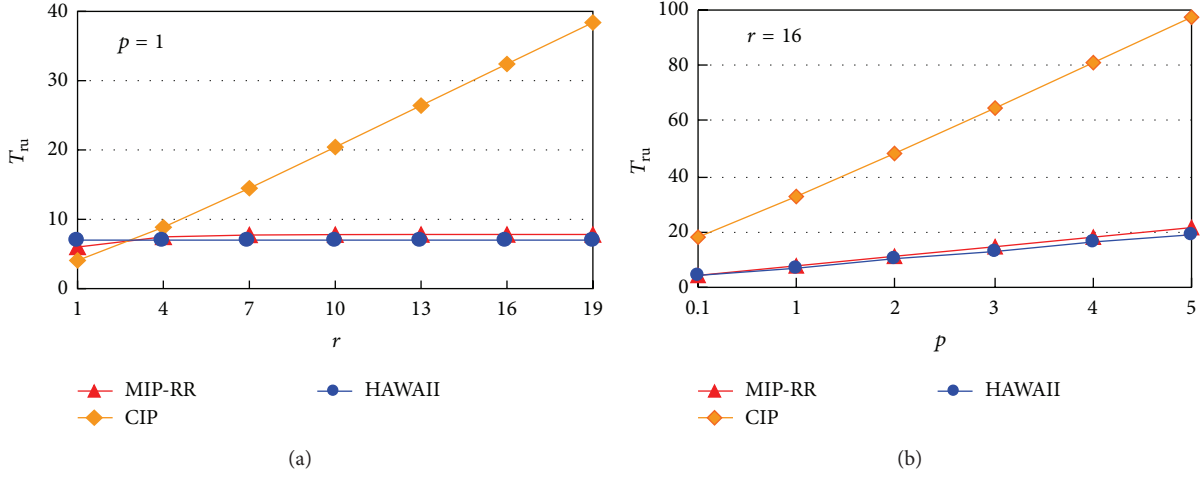


FIGURE 3: Performance of different micromobility protocols.

If MH handoff happens within the i th layer, assume that cross MRA is a BS at the j th layer, and then route update time is

$$T_{\text{MIP-RR}(i,j)} = [2(i-j+1) + (2(i-j)+1)p]T. \quad (12)$$

The probability of cross MRA in the j th layer cell can be obtained by the administration domain structure as follows:

$$P_{(i,j)} = \begin{cases} \frac{1}{2^{i-1}} & j=0 \\ \frac{1}{2^{i-j}} & j=1, 2, \dots, i-1. \end{cases} \quad (13)$$

Then we got the route update time in case of MH handoff happens within the i th layer which is

$$\begin{aligned} T_{\text{MIP-RR}(i)} &= \sum_{j=0}^{i-1} T_{\text{MIP-RR}(i,j)} * P_{(i,j)} \\ &= \left[6 - \frac{4}{2^i} + \left(5 - \frac{4}{2^i} \right) p \right] T. \end{aligned} \quad (14)$$

According to formula (11), (14), and (3), we could get the route update time when MH is in the i th layer cell of the administration domain:

$$\begin{aligned} T_{\text{MIP-RR}(i)} &= \sum_j P_{i,j} * T_{\text{MIP-RR}(i,j)} \\ &= \left[\frac{22}{5} - \frac{8}{(5 * 2^i)} + \left(\frac{17}{5} - \frac{8}{(5 * 2^i)} \right) p \right] T. \end{aligned} \quad (15)$$

In the case of CIP protocol, route update packets from MH were sent to GW via new BS when handoff happens but reply packets are not need, so route update time is only related to the distance between new BS and GW. Route update time when MH handoff happens from the i th layer cell to the j th layer ($j = i-1, i, i+1$) cell is as follows:

$$T_{\text{CIP}(i,j)} = (j+1)(1+p)T. \quad (16)$$

Similarly, the route update time of CIP protocol when MH is in the i th layer cell of the administration domain is

$$T_{\text{CIP}(i)} = \sum_j P_{i,j} * T_{\text{CIP}(i,j)} = \left(i + \frac{6}{5} \right) (1+p)T. \quad (17)$$

In the case of HAWAII protocol, route update packets from MH were sent to the old BS via new BS when handoff happens and the reply packets along the opposite path return to MH. Because the old BS and the new BS are always directly connected to each other in all network models, route update time of HAWAII protocol when handoff happens has nothing to do with MH's location:

$$T_{\text{HAWAII}} = (4+3p)T. \quad (18)$$

As to formula (15) and (17), if we replace i with d which is the average distance from the cell that MH locates to GW, we would get the average update time of MIP-RR protocol and CIP protocol. We take $T = 1$; the average route update time values (T_{ru}) of different micromobility protocols when r and p take different values, respectively, are shown in Figure 3.

Figure 3 shows that, with the increase of r , the CIP protocol route update time increases sharply; MIP-RR protocol route update time slowly increases and converges to a fixed value while HAWAII protocol path update time remains the same. With the increase of p , every micromobility protocol route update time shows a linear growth. The slope of CIP protocol is greater and increases with the increase of r while the slope of MIP-RR protocol and HAWAII protocol is smaller and is less affected or not affected by r .

CIP protocol route update time is much larger than that of MIP-RR protocol and HAWAII protocol in most cases; this is because CIP protocol route update packets need to be sent to GW continuously, so this route update method is not advisable. Since we used the adjacent BS, all interconnect structure network model, making HAWAII protocol work in the best environment, so HAWAII protocol route update time is shortest in Figure 3, but for other network structure (such as strict tree structure) HAWAII protocol route update

time will increase. From Figure 3, we observed that MIP-RR protocol route update time is very close to the optimal value of that of HAWAII protocol and has a clear upper bound, which is a very nice feature that can ensure the maximum delay of handoff. Therefore the route update method of MIP-RR protocol has the best performance.

4. Mobility Management Cost

Broadly speaking, mobility management cost refers to all of the costs of supporting MH mobility, including the terminals and mobility support nodes processing costs and the bandwidth cost of the network and position database storage overhead. This paper mainly focused on the signaling overhead brought to network in order to support MHs' mobility which is an important performance indicator when measuring mobility support protocols' performance.

In the sections below, we compared the signaling overhead when only using Mobile IP with introducing hierarchical mobility and when applying different micromobility protocols. First we present the following definition.

- (1) The signaling overhead of route update packets transmission in administration domain cable link in bytes equals the value of packet size multiply link distance in hop count.
- (2) Because of the limit of wireless link bandwidth resources, the signaling overhead of route update packets transmission in the wireless link is α times more than that of cable link.
- (3) Assume the distance from GW to the HA of MH is w hop and the signaling overhead of route update packets transmission in the wireless link is β times more than that of cable link, for WAN bandwidth is expensive, too.
- (4) Denote the handoff frequency of MH by F_{HO} .

4.1. Comparison of Mobile IP with Micromobility Protocols.

In the case of using Mobile IP, route update packets from MH were passed to HA through GW; the reply packets along the opposite path return to MH. In order to maintain path information, MH need periodically to send refresh message to HA (use route update packet to refresh.). We denote refresh message sending frequency by F_{RN} and the size of route update packet and reply packet by R_{update} and R_{reply} , respectively. The signaling overhead brought to network in order to support MHs' mobility using Mobile IP is

$$C_{MIP} = (\alpha + D' + \beta w)(R_{update} + R_{reply})(F_{HO} + F_{RN}). \quad (19)$$

In the formula above, D' denotes the average distance between new BS and GW; from formula (2) and (3), we could get D' :

$$D' = \sum_{i,j} (D + j - i) * p_{i,j} = D + \frac{1}{5}. \quad (20)$$

In the case of introducing hierarchical mobility, route update packets from MH were passed to HA when handoff happens, and the signaling overhead brought to network is the same as that of using Mobile IP. As to intradomain handoff, in the worst cases, path update packets from MH are sent to the administration domain GW and reply packets along the opposite path are to return to MH. In order to maintain path information, MH need periodically to send refresh message to HA. The signaling overhead brought to network when interdomain handoff and intradomain handoff happen once can be formulated as follows, respectively:

$$C_{inter} = (\alpha + D' + \beta w)(R_{update} + R_{reply}), \quad (21)$$

$$C_{intra} = (\alpha + D')(R_{update} + R_{reply}).$$

From formula (1), the probability of MH interdomain handoff can be calculated as follows:

$$p_{inter} = P_R * P_{R,R+1} = \frac{2^R}{(5 * 2^R - 4)}. \quad (22)$$

So the signaling overhead brought to network in order to support MHs' mobility introducing hierarchical mobility is

$$\begin{aligned} C_{HM} &= [p_{inter} * C_{inter} + (1 - p_{inter}) * C_{intra}] \\ &\quad * F_{HO} + C_{inter} F_{RN} \\ &= [(\alpha + D' + \beta w p_{inter}) F_{HO} + (\alpha + D' + \beta w) F_{RN}] \\ &\quad \cdot (R_{update} + R_{reply}). \end{aligned} \quad (23)$$

The difference of the signaling overhead brought to network in order to support MHs' mobility when only using Mobile IP with introducing hierarchical mobility is calculated as follows:

$$C = C_{MIP} - C_{HM} = (1 - p_{inter}) \beta w F_{HO} (R_{update} + R_{reply}). \quad (24)$$

Because $p_{inter} < 1$, so $C > 0$, which means that the signaling overhead when only using Mobile IP is larger than that of introducing hierarchical mobility in any case. In order to study the advantage of introducing hierarchical mobility and micromobility protocols, we take $\beta = 2$, $F_{HO} = 0.25$ times/s, and $R_{update} = R_{reply} = 60$ bytes and calculated out different w values and C values when R takes different values. The results are shown in Figure 4.

With the increase of the administration domain radius R , the difference of the signaling overhead brought to network in order to support MHs' mobility when only using Mobile IP with introducing hierarchical mobility C also increases, but the increase rate decreases gradually, and finally the difference converges to a fixed value (under the situation of $w = 10, 50$, and 100 , the fixed value is $480, 2400$, and 4800 , resp.). This suggests that when the administration domain is not big, the bigger the R is, the more the advantages of

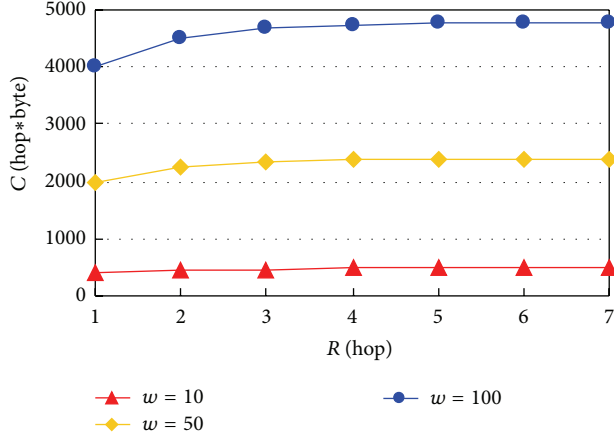


FIGURE 4: Signaling overhead of Mobile IP and micromobility protocols.

introducing hierarchical mobility are. But in the case of administration domain being very large, this advantage is not shown. In addition, as the increase of the distance between administration domain GW and HA, C shows a linear growth, which suggests that the further away MH from the local network is, the more advantages introducing hierarchical mobility shows.

4.2. Comparison of Different Micromobility Protocols. In the case of the comparison of the signaling overhead brought to network in order to support MHs' mobility when applying different micromobility protocols such as MIP-RR, CIP, and HAWAII, because all of them adopted Mobile IP in inter-domain handoff situation, we just discuss the comparison in intradomain situation.

Suppose MH is in an i th layer cell of the administration domain before handoff ($i = 1, 2, \dots, R - 1$); the packet forwarding path of MH within domain is the optimal path from GW to the current BS. There are three cases that may occur (Figure 5), and the handoff probability of the various situations is described by formula (3).

In the case of MIP-RR protocol, the signaling overhead brought to network in order to support MHs' mobility comprises two parts: one part is brought by Mobile IP Regional Registration request packets and reply packets which are used to establish new route. Request packets from MH were sent to GW via new BS and ending at cross MRA (the lowest level of public MRA on both new route and old route of MH). The reply packets along the opposite path return to MH. The other part is brought by binding update packets and reply packets which are used to delete the old route; binding update packets are sent to the old BS by the new BS; then the old BS forward the packets to GW direction and use reply packets to respond layer by layer until cross MRA; cross MRA send reply packets to MH at last. We denote the size of regional registration request packets and reply packets and binding update packets and reply packets by R_{RR} , R_{RP} , R_{BU} , and R_{BA} , respectively.

If MH handoff happens from the i th layer to the $(i - 1)$ th layer (Figure 5(a)), the new BS is cross MRA and the new

BS links directly to the old BS; signaling overhead brought to network when handoff happens is

$$C_{\text{MIP-RR}(i,i-1)} = \alpha (R_{RR} + R_{RP}) F_{\text{HO}} + [2R_{BU} + (\alpha + 1) R_{BA}] F_{\text{HO}}. \quad (25)$$

In the situation that MH handoff happens from the i th layer to the $(i + 1)$ th layer (Figure 5(b)), the old BS is cross MRA and the new BS links directly to the old BS; signaling overhead brought to network when handoff happens is

$$C_{\text{MIP-RR}(i,i+1)} = (\alpha + 1) (R_{RR} + R_{RP}) F_{\text{HO}} + [R_{BU} + (\alpha + 1) R_{BA}] F_{\text{HO}}. \quad (26)$$

If MH handoff happens within the i th layer, assume that cross MRA is a BS at the j th layer cell ($j = i - 1, i - 2, \dots, 0$); signaling overhead brought to network when handoff happens is

$$C_{\text{MIP-RR}(i,i)j} = (\alpha + i - j) (R_{RR} + R_{RP}) F_{\text{HO}} + [(i - j + 1) R_{BU} + (\alpha + 2(i - j)) R_{BA}] F_{\text{HO}}. \quad (27)$$

The probability of cross MRA in the j th layer cell can be obtained by the administration domain structure as follows:

$$P_{(i,i)j} = \begin{cases} \frac{1}{2^{i-1}} & j = 0 \\ \frac{1}{2^{i-j}} & j = 1, 2, \dots, i - 1. \end{cases} \quad (28)$$

According to formula (27) and (28), we could get the signaling overhead brought to network when MH handoff happens within the i th layer:

$$C_{\text{MIP-RR}(i,i)} = \sum_{j=0}^{i-1} C_{\text{MIP-RR}(i,i)j} * P_{(i,i)j}. \quad (29)$$

We got the signaling overhead brought to network when MH handoff happens in the situation that MH is located in an i th layer cell of the administration domain and uses the MIP-RR protocol from formula (25), (26), (29), and (3):

$$C_{\text{MIP-RR}(i)} = \frac{1}{5} C_{\text{MIP-RR}(i,i-1)} + \frac{2}{5} C_{\text{MIP-RR}(i,i+1)} + \frac{2}{5} C_{\text{MIP-RR}(i,i)}. \quad (30)$$

MIP-RR protocol sends out route delete packets to delete the old route explicitly. But CIP protocol uses a very different route maintenance strategy; the old route would not be deleted explicitly but would be automatically deleted due to timeout; for this reason, MH must periodically send refresh packets to prevent the route information from being deleted automatically. As to CIP, though MH must send update packets and refresh packets to GW when every handoff happens, MH uses data packets instead of update packets and refresh packets to update and refresh the route when MH

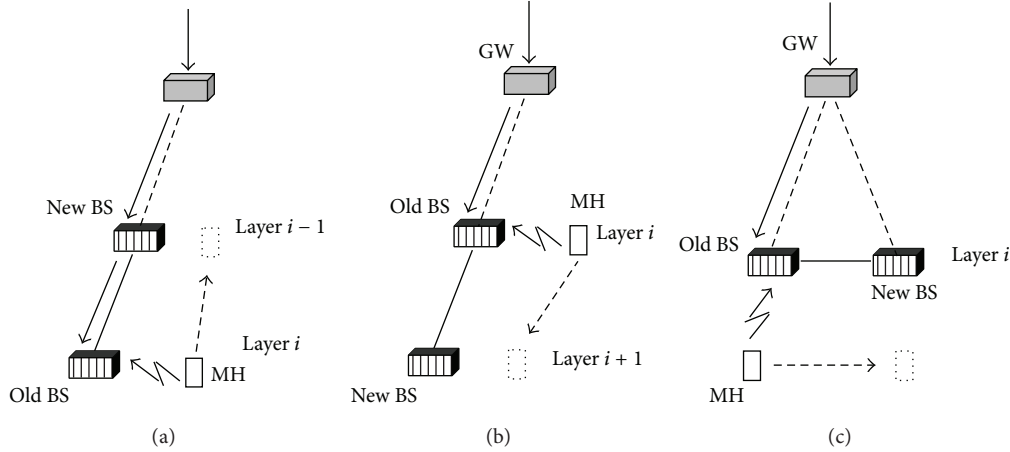


FIGURE 5: Three cases for MH handoff.

need to send out data to reduce network signaling overhead. Set the probability of the need of sending update packets and refresh packets to update and refresh the route as p , the size of route update packet as R_{RU} , descending packet rate as ν , and path information timeout time and refresh cycle ratio as γ . The literature [13] calculated the best refresh cycle as follows:

$$T_{RU} = \sqrt{\frac{pR_{RU}}{[(\gamma - 1/2)\nu F_{HO}]}}. \quad (31)$$

In the case of CIP protocol, route update packets from MH were sent to GW via new BS when handoff happens but reply packets are not needed, so the signaling overhead brought to network in order to support MHs' mobility is only related to the distance between new BS and GW. The signaling overheads brought to network by refreshing periodically when MH handoff happens from the i th layer cell to the $(i - 1)$ th layer, the $(i + 1)$ th layer, and the i th layer cell are shown as follows:

$$\begin{aligned} C_{CIP(i,i-1)} &= p(\alpha + i - 1)R_{RU}F_{HO}, \\ C_{CIP(i,i+1)} &= p(\alpha + i + 1)R_{RU}F_{HO}, \\ C_{CIP(i,i)} &= p(\alpha + i)R_{RU}F_{HO}, \\ C_{CIP(i)Refresh} &= \frac{p(\alpha + i)R_{RU}}{T_{RU}}. \end{aligned} \quad (32)$$

According to formula (32) and (3), we could get the signaling overhead brought to network when MH handoff happens in the i th layer cell of the administration domain:

$$\begin{aligned} C_{CIP(i)} &= \frac{1}{5}C_{CIP(i,i-1)} + \frac{2}{5}C_{CIP(i,i+1)} + \frac{2}{5}C_{CIP(i,i)} \\ &+ C_{CIP(i)Refresh}. \end{aligned} \quad (33)$$

In the case of HAWAII protocol, route establishing packets from MH were sent to the old BS via new BS when handoff happens and the reply packets along the opposite path return to MH. This process completes the establishment

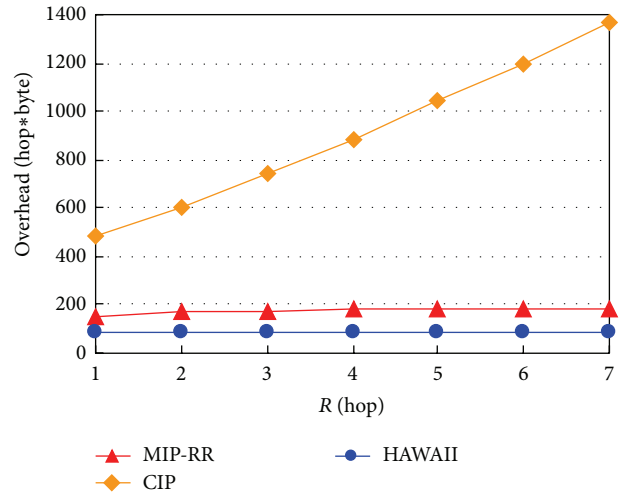


FIGURE 6: Signaling overhead of different micromobility protocols.

of the new route information and the deletion of the old route information. Set the size of route establishing packet as R_{PS} . Because the new BS is always directly connected to the old BS, so the signaling overhead brought to network has nothing to do with the position of MH; it is

$$C_{HAWAII} = 2(\alpha + 1)R_{PS}F_{HO}. \quad (34)$$

As to the formula above, if we replace i with D which is the average distance from the cell that MH locates to GW, and take $\alpha = 2$, $F_{HO} = 0.25$ times/s, $p = 0.1$, $\gamma = 3$, $\nu = 64$ kbps, all the route update packets (including reply packets) have the same size of 60 bytes. We could calculate the signaling overhead brought to network in order to support MHs' mobility, respectively, with administration domain radius R changing when applying various mobility support protocols. The results are shown in Figure 6.

Figure 6 shows that CIP protocol signaling overhead brought to network in order to support MHs' mobility increases with the increase of R , and much greater than other protocols. This phenomenon is determined by CIP protocol

route maintenance strategy (frequently sends periodic refresh packets) and route update method (route update packets sent to GW rather than cross MRA). Though MH uses data packets instead of update packets and refresh packets to update and refresh the route when MH need to send out data to reduce network signaling overhead, even if 90% of the route update packets can be piggybacked by data packet ($p = 0.1$), the signaling overhead is also quite large (especially administration domain scope in a larger case), so CIP protocol route maintenance strategy and route update method are not desirable.

We can find that MIP-RR protocol signaling overhead brought to network in order to support MHs' mobility slightly increased with the increase of R and with a clear upper bound. However HAWAII protocol signaling overhead brought to network in order to support MHs' mobility is unaffected. In the case of the same administration domain size, MIP-RR protocol signaling overhead is slightly larger than that of HAWAII protocol. HAWAII protocol uses the same route maintenance strategy (delete the old route explicitly) as MIP-RR protocol, but MIP-RR protocol route update method sends route update packets to GW via new BS and ending at cross MRA while in HAWAII protocol route update packets were sent to the old BS via the new BS. For the latter may cause nonoptimal routing, therefore MIP-RR protocol path update method is more worth to recommend.

5. Conclusion

This paper makes a study of mobility support protocols network layer handoff delay and analysis focused on the route update time of mobility support protocols. The result shows that route update time is relate to the route update methods of mobility support protocols. Micromobility protocols route update packets were sent only within administration domain, so the route update time is far less than that of Mobile IP protocol. As to different micromobility protocols, CIP protocol route update packets should be sent to GW continuously; route update time is larger. HAWAII protocol route update packets were sent to the old BS directly; route update time was greatly influenced by network structure. The route update method of MIP-RR protocol has an optimal performance, because the route update packets were sent to the GW direction and ended at the cross MRA. MIP-RR represents the development direction of micromobility protocols.

Conflict of Interests

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

Acknowledgments

This work was supported by the Prospective Research Project on Future Networks of Jiangsu Future Networks Innovation Institute (no. BY2013095-1-16) and the National Key Technology R&D Program of China (no. 2013BAK06B03).

References

- [1] A. Myles, D. B. Johnson, and C. Perkins, "A mobile host protocol supporting route optimization and authentication," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, pp. 839–849, 1995.
- [2] C. Perkins, "IP mobility support for IPv4, revised," Tech. Rep. IETF RFC5944, 2010.
- [3] C. Perkins, D. Johnson, and J. Arkko, "Mobility Support in IPv6," IETF RFC6275, 2011.
- [4] E. Fogelstroem, A. Jonsson, and C. Perkins, "Mobile IPv4 regional registration," IETF RFC 4857, Internet Engineering Task Force, Dallas, Tex, USA, 2007.
- [5] I.-R. Chen, W. He, and B. Gu, "Proxy-based regional registration for integrated mobility and service management in mobile IP systems," *Computer Journal*, vol. 50, no. 3, pp. 281–293, 2007.
- [6] S. Omer, S. Qamar, R. Vesilo, and E. Dutkiewicz, "Efficient mobility management using simplified cellular IP," in *Proceedings of the Australasian Telecommunication Networks and Applications Conference (ATNAC '13)*, pp. 13–18, November 2013.
- [7] I. V. Hernández and Q. E. E. Morones, "Performance analysis of the cellular IP mobility protocol," *IEEE Latin America Transactions*, vol. 5, no. 2, pp. 99–102, 2007.
- [8] R. Ramjee, K. Varadhan, L. Salgarelli, S. R. Thuel, S.-Y. Wang, and T. La Porta, "HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 396–410, 2002.
- [9] A. Wali, M. A. Ghazali, and S. R. Ayyubi, "Comparative analysis of mobile IP and HAWAII," *Communications in Computer and Information Science*, vol. 20, pp. 169–179, 2008.
- [10] A. T. Campbell, J. Gomez, S. Kim, C.-Y. Wan, Z. R. Turanyi, and A. G. Valko, "Comparison of IP micromobility protocols," *IEEE Wireless Communications*, vol. 9, no. 1, pp. 72–82, 2002.
- [11] K. Pei, J.-D. Li, and F. Guo, "Mobile IP routing optimization performance analysis and simulation," *Journal of Electronics*, vol. 30, 2002.
- [12] T. Hong, X. S. Min, W. Z. Fu, and Z. Jun, "Improved mobile IP register performance analysis," *Journal of Communication*, vol. 26, no. 6, 2002.
- [13] A.-Q. Zhao, "Investigation on handoff performance of mobility support protocols," *Journal of Software*, vol. 16, no. 4, pp. 587–594, 2005.
- [14] L. Piroaska, A. M. Ronai, and R. Z. Turanyi, "Cost of location maintenance related signaling in IP micro mobility networks," in *Proceedings of the Transcom Conference*, Zilina, Slovakia, June 2001.
- [15] T. Pagtzis and C. Perkins, "Performance issues for localized IP mobility management," in *Proceedings of the IEEE International Conference on Networks (ICON '02)*, August 2002.
- [16] J.-H. Lee, T. Ernst, and T.-M. Chung, "Cost analysis of IP mobility management protocols for consumer mobile devices," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 1010–1017, 2010.
- [17] M. Song and J. Jeong, "Performance analysis of a novel inter-networking architecture for cost-effective mobility management support," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 4, pp. 1344–1367, 2014.
- [18] A.-Q. Zhao, "Research on scalability of mobility support protocols," *Journal of Harbin Institute of Technology*, vol. 38, no. 10, pp. 1732–1735, 2006.

Research Article

Trust-Aware and Low Energy Consumption Security Topology Protocol of Wireless Sensor Network

Zuo Chen,^{1,2} Min He,¹ Wei Liang,³ and Kai Chen¹

¹ College of Computer Science and Electronic Engineering, Hunan University, Changsha, Hunan 410082, China

² Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

³ School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

Correspondence should be addressed to Wei Liang; [idlink@163.com](mailto:ldlink@163.com)

Received 18 August 2014; Accepted 8 October 2014

Academic Editor: Fei Yu

Copyright © 2015 Zuo Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor network (WSN) is a kind of distributed and self-organizing networks, in which the sensor nodes have limited communication bandwidth, memory, and limited energy. The topology construction of this network is usually vulnerable when attacked by malicious nodes. Besides, excessive energy consumption is a problem that can not be ignored. Therefore, this paper proposes a secure topology protocol of WSN which is trust-aware and of low energy consumption, called TLES. The TLES considers the trust value as an important factor affecting the behavior of node. In detail, the TLES would take trust value, residual energy of the nodes, and node density into consideration when selecting cluster head nodes. Then, TLES constructs these cluster head nodes by choosing the next hop node according to distance to base station (BS), nodes' degrees, and residual energy, so as to establish a safe, reliable, and energy saving network. Experimental results show that the algorithm can effectively isolate the malicious node in the network and reduce the consumption of energy of the whole network.

1. Introduction

With the development of wireless communications, electronics, and sensing technology, the wireless sensor networks (WSN) [1] have attracted much attention. WSN consist of many wireless sensors which have sensing, data processing, and short distance wireless communication function. These embedded sensors could be self-organizing and work together to sense and collect all kinds of interesting environment data. Moreover, they also analyze and process the original data to obtain accurate information under various environmental conditions [2]. The excellent characteristics of WSN make it have a broad application prospect in military defense, environmental monitoring, biological, medical, disaster relief, and commercial applications, and so forth [3–5].

In general, the WSN nodes are equipped with independent battery and usually deployed of large numbers in the wild places where people almost could not reach. It is an impossible mission to recharge or replace the sensor battery. In order to reduce the energy consumption, the communication radius of node is strictly limited. The topology protocols

of WSN commonly focus on how to separate the whole network into clusters and how to make multihops construction among these cluster heads for transferring sensor data to base station by self-organization. In the open, distributed, and dynamic environment, the construction of network topology is vulnerable, which may lead entire network to be unsafe. For WSN, how to ensure the security of the communication is an important issue in the process of constructing network topology [6, 7].

In recent years, people have put forward different security routing protocols. Most of these are based on the traditional security mechanisms of the cryptosystems, which need much more memory and energy consumption. The wireless sensor network is composed of many small sensor nodes with limited bandwidth and stringent node constraints in terms of power and memory. What is more, the cryptosystems can only resist external attack; once internal nodes have mutations or attacks, they will not be able to be identified. So the traditional cryptosystems of traditional security mechanisms are not fully applicable to wireless sensor networks [8].

To solve the above problems, researchers have proposed the trust management mechanism. Trust is defined as the binary relation occurring in subject and object. The trust management mechanism depends on the history record of object behavior or interaction behavior. The record is used to calculate a trust value. The trust value provides a prediction of future behavior and determines the object's next step. The evaluation of the trust value includes node trust, link trust, and service trust. This strategy makes the trust management mechanism effective in improving the security of the network in an open environment.

This paper tries to take node trust into consideration when building a network topology, so as to ensure the security of the network communication. The TLES algorithm is based on the analysis of node behavior. It develops a variety of trust factors and then performs comprehensive analysis with direct information and recommended information. This working principle can dynamically reflect changes in the trust value between the nodes. TLES combines trust value, residual energy, and density together. This algorithm uses the local optimum principle to choose the cluster head node. Cluster head selects the next hop based on the residual energy, distance to BS, and degree of other cluster heads. As a result, it can effectively eliminate the malicious nodes in network and achieve safe, rational node communication. Besides, the energy consumption of the network is reduced.

The rest of this paper organization is as follows: the second part is a brief review of the related research work; the third part presents the system model and the problem description; the fourth part shows the details of the topology algorithm; the fifth part is about simulation results and the analysis; the last part is conclusions for summary and future work.

2. Related Work

There have been many researches on WSN trust models. Zhan et al. proposed a plane routing protocol based on trust, which is called TARP. The TARP uses the trust value and energy cost to decide the routing path. This protocol can prevent malicious nodes from tampering with routing information and misleading network traffic [9]. Raje and Sakhare [10] proposed a mechanism of cluster head election. This mechanism is based on trust model and uses a certain probability to choose cluster head. Other ordinary nodes would join a cluster head after analyzing energy and the cluster head's trust values. If no node joins a cluster head, the cluster head would become a common node. Repeating the above process until all common nodes have found their cluster heads, which is complex calculation process that would bring network too much burden and energy cost, Crosby et al. proposed a cluster head election algorithm based on trust value. In this algorithm, neighbor nodes monitor data packets and control packets forwarding information, calculate the trust value, and select the neighbor node with the highest trust value as the cluster head. This algorithm combines challenge response with redundancy strategy to reduce the possibility of malicious nodes becoming cluster head [11]. In [12], Safa et al. proposed a hierarchical routing

algorithm which is called CBTRP. For the CBTRP, neighbor nodes self-organize into a cluster structure according to the corresponding trust value. To ensure the safety of data transmission, the CBTRP would send data to trust cluster head directly and apply directed diffusion. Heinzelman et al. proposed an improved LEACH algorithm [13] based on trust value, that is, LEACH-TM [14]. The LEACH-TM algorithm uses trust value to optimize the selection of cluster head and the formation of the cluster structure. In this way, the LEACH-TM can identify the malicious nodes, reduce data packet loss, and enhance network security. There is a TARP [15] protocol which applies a trust-based routing scheme responsible for routing messages from the different nodes to the base station. It is based on idea of node cooperation which forwards the neighbor messages. It uses the concept of cooperation in terms of routing reputation. TARP achieves significant improvements in terms of energy consumption and scalability. This protocol exploits nodes' past routing behavior and link quality to determine efficient paths, but it does not offer protection against the identity deception through replaying routing information.

These above methods mainly focus on single network security threats without considering trust value across the board; thus it may ignore security and performance defects of the trust routing itself. For example, the computation of trust value is too complex, malicious nodes are difficult to identify, and key nodes are vulnerable. Therefore, this paper proposes a secure routing algorithm based on trust for wireless sensor network (TLES). The TLES synthesizes direct and recommended information for trust calculation. So it can dynamically reflect the change of trust value between nodes. Besides, it takes trust value, energy cost, and node density into consideration. The nodes compete and select a cluster head. The cluster head node chooses the next-hop node according to energy cost, distance, and degree. Using this strategy, the TLES can effectively eliminate the malicious nodes in the network. It can also ensure security and rationality of node communication effectively, as well as reducing network energy cost.

3. System Model and Problem Description

This paper proposes TLES, which lets node construct the topology structure of the whole network according to the neighbor node's trust value, residual energy, and distance to base station. Models and problems of TLES topology construction are described as follows.

3.1. Network Model. Supposing N sensor nodes are randomly distributed in the $M * M$ region, the main characteristics of sensor nodes are as follows:

- (1) all sensor nodes have the same initial trust value, energy value, and status;
- (2) there is only one BS node in the WSN, and the BS node's energy is infinite;
- (3) once a sensor node has been deployed, it cannot be moved;

- (4) node is not equipped with GPS, but each node can know the location information of the current node;
- (5) a sensor node has many energy levels, so the sensor nodes can dynamically adjust the model of the energy according to the transmission distance.

The first to fourth are the basic properties of wireless sensor networks, and the fifth property is defined energy levels for the communication within the cluster and the communication between clusters; the two communication modes have different energy consumption.

3.2. Wireless Communication Model. This paper uses the same wireless communication model in [16]. How to calculate d_o is shown in (1). If $d \leq d_o$, the node energy consumption is proportional to the square of the communication distance; if $d > d_o$, the node energy consumption is proportional to the biquadrate of the communication distance. The above two models are called the free space model (free space) and multipath fading model (multipath fading), respectively. In order to realize that nodes' energy consumption have proportional relationship to the square of the distance, the broadcast distance of the nodes and the communication distance R were set to d_o in this paper. The energy consumption of sending k -bits data is as shown in formula (2). Consider

$$d_o = \sqrt{\frac{E_{fs}}{E_{mp}}} \quad (1)$$

$$E_{tr}(k) = E_{elc}(k) + E_{amp}(k, d) = \begin{cases} k \times E_{elc} + k \times E_{fs} \times d^2; & \text{if } d \leq d_o \\ k \times E_{elc} + k \times E_{mp} \times d^4; & \text{if } d > d_o. \end{cases} \quad (2)$$

The energy consumption of sensor nodes receiving k -bits data is as shown in

$$E_{rx}(k) = k \times E_{elc}. \quad (3)$$

E_{elc} stands for the energy consumption when receiving and sending 1 bit data, E_{amp} represents the energy consumption when node fuse 1 bit data, E_{fs} stands for the consumption of energy when sending 1 bit data in the free space model, and E_{mp} represents the consumption of energy of sending 1 bit data in the multipath fading model.

3.3. Problem Description. In order to solve these weaknesses existing in the previous studies, TLES protocol needs to meet the conditions as follows:

- (1) the network node communication radius is less than or equal to the d_o ; therefore all the nodes in the network could meet the free space model which can effectively reduce the energy consumption;
- (2) it is difficult for nodes to obtain global information, with the increasing scale of WSN; the node should construct the whole network topology only by local neighbor nodes' information;

- (3) the node's trust value is dynamic, the changes of which should be able to accurately reflect the node security;
- (4) in TLES algorithm, all nodes try their best to deliver packets to their next node, integrating a variety of trust mechanisms to select a neighbor node with the highest trust value as the cluster head node;
- (5) communication between cluster head nodes should try to satisfy the free space model; the communication radius is less than or equal to the d_o .

4. Details of TLES

TLES algorithm consists of two parts. The first part is to calculate the trust value of nodes and select cluster head nodes according to the trust value, residual energy, and the density of nodes. If the ordinary node's trust value is less than a certain threshold, it could not be allowed to join any cluster heads. The second part is to build a weighted tree. All the cluster head nodes select the next-hop nodes, according to the node information including the value of residual energy, the distance between cluster and BS, and the value of clusters' degree, so as to construct the whole network topology and transmit the information to the BS node finally.

4.1. The Calculation of Trust Value. Trust depends on the subject's (evaluating node) assessment to the object (evaluated node) and the recommendation of other nodes, and the value will change according to object's behavior. Considering the characteristic of self-organizing and multiple hops in wireless sensor network, the trust evaluation mechanism should be set up with no core node. Nodes monitor each other's behavior between neighbors, and use the direct and indirect trust value to get comprehensive trust values.

(1) *Sending Rate Factor* $SF_{i,j}(t)$. Evaluating node i monitors the quantity sending of the evaluated node j . If the number is lower than the lower limit threshold T_L , the node can be regarded as a selfish node. If the number is more than the upper limit threshold T_H , the node may have performed attack as behavior of denial of service. The sending rate factor's formula is shown as follows:

$$SF_{i,j}(t) = \begin{cases} \frac{SP_{i,j}(t) - T_L}{ES_{i,j}(t) - T_L} & SF_{i,j}(t) \leq ES_{i,j}(t) \\ \frac{T_H - SP_{i,j}(t)}{T_H - ES_{i,j}(t)} & SF_{i,j}(t) > ES_{i,j}(t). \end{cases} \quad (4)$$

In formula (4), $SP_{i,j}(t)$ stands for the quantity sending of the period t and $ES_{i,j}(t)$ represents the expected value of the quantity sending of the period t . When $T_L = 300$, $T_H = 700$, and $ES_{i,j}(t) = 500$, the changes of $SF_{i,j}(t)$ are shown in Figure 1.

It is clear that the range of $SF_{i,j}(t)$ is from 0 to 1. If the value of $SP_{i,j}(t)$ is closer to $ES_{i,j}(t)$, the value of $SF_{i,j}(t)$ is closer to 1, which means that the node gets a relatively higher trust value.

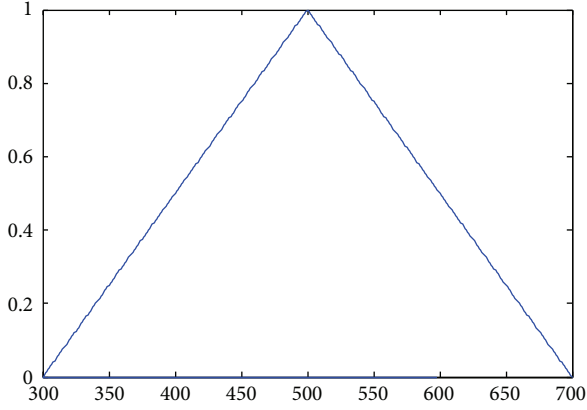


FIGURE 1: The variation of sending rate factor.

(2) *Consistency Factor* $CF_{i,j}(t)$. To prevent malicious nodes forged packets, we need to compare the data of node collecting by itself with the data collecting by neighbor nodes, which is called the analysis of spatial coherence. The data gathered from different nodes in the same local area of local networks generally shows a high degree of correlation. The evaluating node i monitors the packet of evaluated node j , compared with its acquisition data. The evaluating node i monitors the packet of evaluated node j , and i compares the data collecting by itself with the data collecting by j . If the difference between the two data is within a certain range, the evaluating node i and evaluated node j have a consistent opinion about the monitored environment. Consider

$$CF_{i,j}(t) = \frac{CP_{i,j}(t)}{CP_{i,j}(t) + NCP_{i,j}(t)}. \quad (5)$$

In formula (5), $CP_{i,j}(t)$ stands for the number of nodes having the same packet with evaluated node j and $NCP_{i,j}(t)$ stands for the number of inconsistent packet. $CP_{i,j}(t) + NCP_{i,j}(t)$ is the number of all the packet that i received from its surrounding nodes.

(3) *Packet Loss Rate Factor* $DF_{i,j}(t)$. Because the energy of node is limited in WSN, some nodes cannot communicate with the base station (BS) directly and need other nodes as relay node to forward information to the BS by a multiple-hop topology. The packet drop is likely to exist in the process of transmission, which leads to the loss of information. The formula of packet loss rate factor is shown in

$$DF_{i,j}(t) = \frac{T(t)}{R(t)}. \quad (6)$$

In formula (6), $T(t)$ is the amount of all the packets sent by all the nodes, in the period t . $R(t)$ is the amount of packet received by all the nodes times, at the same time. Obviously, the range of $DF_{i,j}(t)$ is also from 0 to 1.

Suppose that nodes i and j are neighbor nodes. When node i assesses node j , considering the attack of malicious nodes and selfish node, we need to combine all trust factors mentioned previously. Firstly, calculating nodes' direct trust

value and then calculating the indirect trust value through other node k which is connecting both nodes i and j , the computation formula of direct trust value is as follows:

$$\begin{aligned} Td_{i,j}(t) &= (1 - \alpha) * SF_{i,j}(t) * CF_{i,j}(t) * DF_{i,j}(t) \\ &+ \alpha * Td_{i,j}(t - 1). \end{aligned} \quad (7)$$

In formula (7), $SF_{i,j}(t)$ is sending rate factor, $CF_{i,j}(t)$ is consistency factor, $DF_{i,j}(t)$ is packet loss rate factor, α is a constant coefficient, and the range is from 0 to 1. The range of $Td_{i,j}(t)$ is from 0 to 1. $Td_{i,j}(t)$ is 0, representing that the node is abnormal node, and the node is untrusted, while 1 stands for the fact that the node is normal completely, and the node is trusted. The greater the trust value is, the more credible the node is.

When selecting the next-hop node, each node is subjective to judge whether the next-hop node could be trusted by calculating the trust of the next-hop node. In order to reduce deviation, the indirect trust value also should be considered, and formula is as follows:

$$Tid_{i,j}(t) = f_t(Td_{i,j}(t), Td_{k,j}(t)). \quad (8)$$

In formula (8), $Td_{i,j}(t)$ is the direct trust value of evaluated node j by i and $Td_{k,j}(t)$ is the direct trust value of evaluated node j by k , connected simultaneously with node i and node j . $f_t[\cdot]$ can be determined according to the needs of actual network. It can be set into linear, such that $\alpha * Td_{i,j}(t) + \beta * Td_{k,j}(t)$, and $\alpha + \beta = 1$. The value of α and β can be determined according to the actual needs. If we want to pay more attention to trust value of the other nodes, we can set the β value higher. However, if the node trust value judgment by own is more important, α could be higher.

4.2. *The Selection of Cluster Head.* Before the first choice of cluster, base station nodes globally broadcasted, each node receives the base station's information and calculates the distance between itself and base stations. Then, each node broadcasts information of itself in local area within the range of distance d_o . When other cluster head nodes receive a message, then they will send a confirmation message to the sending node. After each node calculates the p_{ch} , each node will broadcast the p_{ch} of itself, and the range is d_o . Consider

$$p_{ch} = f_p(E_{current}, T_{ch}, S_d) \quad (9)$$

where $E_{current}$ represents the residual energy of node, T_{ch} represents the trust value of nodes, and S_d is the number of neighbor nodes within a radius of $d_o/4$. $f_p[\cdot]$ is the function to computing nodes' p_{ch} . p_{ch} of node is related to residual energy, the trust value, and the density of the node. We hope that the greater the residual energy of nodes, the greater the trust value, and the greater the density of nodes, the higher the probability of cluster head nodes.

Figure 2 is a schematic diagram of a WSN that consisted of five nodes, and the p_{ch} of node 5 is the biggest. By the following steps, the competition of cluster heads will be completed.

```

(1) if round=1
(2)   Bs_str = Receive_Str(msg form BS) //radio strength from BS
(3) endif
(4) if (S(i).energy > 0 && S(i).type! = "U")
(5)   S(i).state = "N";
(6)   broadcast N_MSG within range  $d_o$ ;
(7)   Receive (confirmation message of neighbor node within the range of  $d_o$ );
(8)   calculate  $p_{ch}$ ;
(9)   broadcast PCH_MSG within range  $d_o$ ;
(10)  receive PCH_MSG from others
(11)   $p = \max(p_{ch} \text{ received}, p_{ch} \text{ own})$ 
(12) endif
(13) if (S(i).  $p_{ch} == p$ )
(14)  S(i).state = "C";
(15) endif
(16) if (S(i).state == "C")
(17)  broadcast MC_MSG within range  $d_o$ ;
(18)  waitfor JOIN_MC_MSG;
(19) endif

```

ALGORITHM 1: The pseudocode of elected cluster head.

- (1) all nodes broadcast their information within the scope of the d_o and receive the other nodes' information within the same scope;
- (2) all the nodes calculate their own p_{ch} according to the received information and then broadcast their p_{ch} information, within the scope of d_o ; all nodes receive p_{ch} of other nodes in this scope; as shown in Figure 2, the node 5 got the p_{ch} value of itself and the other nodes 1, 2, 3, and 4;
- (3) compare its own p_{ch} and other p_{ch} values, if its own p_{ch} is the maximum one; the node will become cluster head; as shown in Figure 2, comparing with the value of node 5 by its own and other values of 1, 2, 3, and 4 nodes, node 5 finds that own value is the biggest, so node 5 becomes a cluster head node by competition.

Algorithm 1 is the pseudocode of clusters:

- (1) BS node broadcast information of base station;
- (2) all the nodes are not isolated and the energy is greater than zero; broadcast their information with range of d_o ; at the same time, all the nodes receive the information of the other nodes within the range of d_o , calculating their p_{ch} ; at last, compare their own P_{ch} and the received neighbor node P_{ch} ;
- (3) the node with the maximum P_{ch} becomes cluster head node;
- (4) cluster head nodes wait for the join messages from other nodes.

4.3. Weighted Spanning Tree Generation. In this section, we want to set up a multihop topology among all the cluster head nodes by generating a weighted spanning tree construction. For convenience of discussion, all the nodes mentioned in this section represent cluster head nodes.

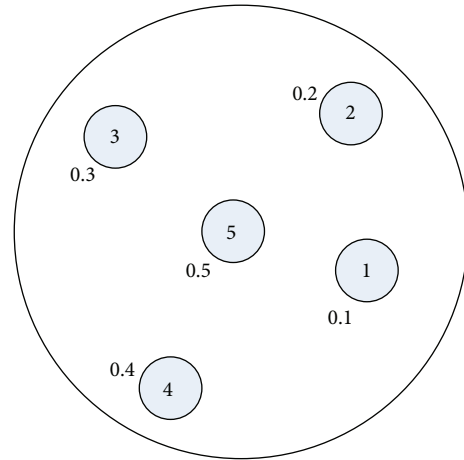


FIGURE 2: The competition of cluster head.

Cluster head selects the next hop from other cluster head nodes, considering the residual energy, the distance between BS and the next node, and degree of the next node. In this paper, the concept of the degree is not just the number of nodes connected directly. For example, the degree of node A is the number of all the nodes, which take the node A as a root node and need node A to forward their information. In TLES, each cluster node broadcasts its information, and the radius of broadcast is d_o . When other cluster nodes have received the message, they will send an acknowledgement message to the node that has broadcasted the message, the acknowledgement message (Message1) and the detail format of the acknowledgement message as shown in Table 2. An acknowledgement message includes the residual energy of the current node, the location of the node, the distance between the node and BS, and the degree of node. When the node has received the acknowledgement message (Message1)

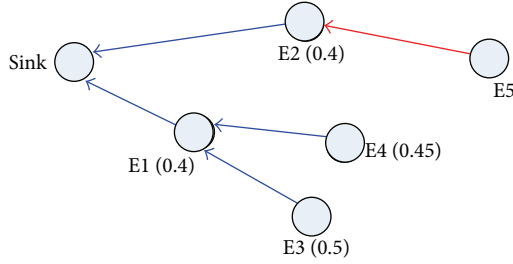


FIGURE 3: Select the next-hop node.

from its neighboring nodes, it will select the next hop from the neighboring nodes.

How to select the next-hop method is as follows.

- (1) Node selects the node whose distance with BS node is less than the distance between the current node and BS node to join the set V from the nodes that have sent Message1 to the current node.
- (2) Select the next-hop node in the set V according to formula (10).
- (3) If node has received the Message1 from the BS node, its next hop is the BS node. Consider

$$\text{Nextnode}(i, j) = a * \frac{\text{distance}(i, j)}{d_o} + b * \frac{E_{\text{current}}}{E_o} + c * \frac{\text{max_degree} - \text{degree}}{\text{max_degree}}. \quad (10)$$

In formula (10), i represents the current node, j represents the neighbor node, and $\text{Nextnode}(i, j)$ represents the link weights between the current node i and node j . $\text{distance}(i, j)$ represents the distance between node i and node j . E_{current} stands for the residual energy of node j . E_o stands for the initial energy. max_degree represents the biggest degree of the previous round of the network. degree represents the degree of j . a , b , and c are constant coefficients, and $a + b + c = 1$. Finally, nodes will select the largest Nextnode as the next hop.

Figure 3 is a schematic diagram of WSN composing 6 nodes. The blue lines are the original connection in the network and red line is the new one connected in this round. Obviously, when choosing the next-hop node, E5 will consider three aspects: distance, energy, and degrees. In this diagram, E1 is closest to the sink node, and E1 is furthest to E5 within the range of d_o (within the range of d_o , sending the data as far as possible regardless of the degree of E1, the E1 can be considered as the next-hop node of E5 temporarily, but, given degree of E1, the E2 has the same energy with E1 whose degree is relatively small. What is more, E2 is the farthest to E5 except for E1, so E2 becomes the next hop of node E5.

In this paper, the concept of the degree is not just the number of nodes connected directly. The degree of one node is the number of all the nodes, which take this node as a root node and need this node to forward information. After

the selection of next cluster head node, the next-hop node's degree should be updated. The formula is as shown in

$$\text{degree}_j = \text{degree}_j + \text{degree}_i. \quad (11)$$

degree_j stands for the current node's degrees and degree_i represents the degree of the next-hop j . All the nodes except for j , which has connected i , need j to forward packet. So, node j should update the value of degree. After node j has updated its degree, it should broadcast the information of the new degree. Then, the next hop of j also should update the one for the change of node j 's degree. This process is repeated until the node has been found, whose next hop is the BS node.

Figure 4 is a schematic diagram of a WSN composing 7 nodes. The blue lines are the original connection in the network and red line is the new one connected in this round. After the E6 has chosen E5 as the next-hop node, E5 needs to update the value of degree, and E4 also needs to update the degree. Because the next hop of E4 is BS node, so no more nodes need to update degree, in this network.

Algorithm 2 is the pseudocode of weighted spanning tree:

- (1) BS node broadcasts its information;
- (2) all the nodes radiobroadcast the information of themselves, and the radiodistance is d_o ;
- (3) nodes receive the radiomessages from the nodes whose distance with the nodes are less than d_o or equal to d_o ;
- (4) all surviving nodes in the network according to the neighbor node information choose the next-hop node, if the distance of node i to BS is greater than that of the node j to the BS and has the Max (Nextnode); the node j will became the next-hop node;
- (5) node j and nodes, whose descendant is j , refresh their degree;
- (6) node i sends its TDMA table to its child nodes, and the connected node receive TDMA.

5. Simulation

The experimental environment is as follows: the area of network is $200 * 200$ and the number of sensor nodes is 200. The initial value of trust of each node is 1. Some malicious nodes are scattered randomly. Malicious nodes may have some bad behaviors, such as packet loss, too big or too small quantity of sending packet, and sending wrong data. Simulation experiment of the initialization parameters are shown in Table 1.

The experiment can be divided into three parts. First, we analyze the detection accuracy of malicious nodes by setting different isolation threshold values. Second, we use the better threshold, gotten by the first part, we analyze the change of average sending ratio, the change of average consistency ratio, and the change of average packet delivery ratio as the change of communication round, in order to verify whether the proposed algorithm could isolate the malicious nodes effectively and improve the average sending ratio, the

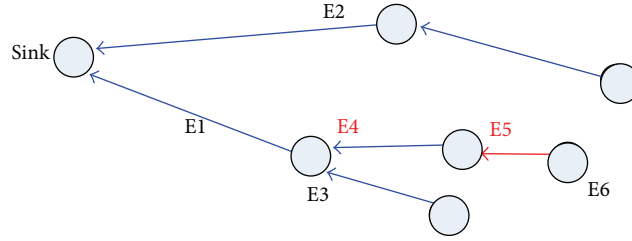


FIGURE 4: Update of degree.

```

(1) if round = 1
(2)   Bs_str = Receive_Str(msg form BS) //radio strength from BS
(3) endif
(4) if (S(i).energy > 0 && S(i).type! = "U")
(5)   if (S(i).distanceToSink < S(j).distanceToSink && S(i).E > 0 && S(j).type! = "U")
(6)     choose max( $a * \text{distance}_{ij}/d_o + b * E_{\text{current}}/E_o + c * (\text{max\_degree} - \text{degree}_{ei})/\text{max\_degree}$ ) as next node
(7)     S(i).nextnode = j;
(8)     degreej += degreeei //updated the next-hop node's degree
(9)     j broadcast N_MSG within range  $d_o$ ;
(10)  end if;
(11) end for
(12) for (All nodes whose residual energy are greater than 0)
(13)   Node I Send TDMA table to it's connected node;
(14)   nodes have connected to the node i receives the TDMA
(15) end for

```

ALGORITHM 2: The pseudocode of weighted spanning tree.

TABLE 1: The initial value of the parameter.

Parameter	Value
E _{tx}	$50e - 9$ J/bit
E _{rx}	$50e - 9$ J/bit
E _{fs}	$10e - 12$ J/bit/m ²
E _{mp}	$0.0013e - 12$ J/bit/m ⁴
EDA	$5e - 9$ J/bit/singal
Control packet length	100 bits
Data packet length	4000 bits
SINK	(0, 0)

TABLE 2: The format of the Message1.

ID	Residual energy	Distance with BS	Location	Degree
----	-----------------	------------------	----------	--------

average consistency ratio, and the average packet delivery ratio of the network. At last, we compare the consumption of energy of TLES with the consumption of energy of some hierarchical routing protocols including LEACH, LEACH, and LEACH_MF.

In Figure 5, the horizontal axis represents the proportion of the malicious nodes (C_p), and the vertical axis represents the percentage of correctly detected malicious nodes in total malicious nodes when the first node dies in the network. Some different curves are gotten by setting different threshold R_o . It can be seen from the diagram that all malicious nodes

can be detected when the threshold R_o is 0.3. Figure 6 shows that average trust values of the malicious nodes are changing following the changing of round number. In this experiment, assuming initial trust value of each node is 1 and then calculating the node trust value according to the previous communication performance of node, we can conclude that, no matter how much the proportion of malicious nodes is, the average trust values are falling and malicious node trust value will be dropped to below 0.3. Therefore, the malicious nodes are detected by setting the isolation threshold value R_o as 0.3 under the experimental environment.s

All the nodes are fully trusted at the beginning of the experiment; that is to say, each node's trust value is 1. The average consistency ratio, the average sending ratio, and the average packet delivery ratio of the whole network are 1. At the beginning of the communication, all the nodes are fully trusted and malicious nodes have not been isolated. Because malicious nodes exist in the network, a lot of abnormal behaviors that include loss of packet, wrong packet, or node not sending packets or sending too much packets will occur, all the three trust factors will decline in the former stage. With the increased rounds of communication, the malicious node will be detected and isolated slowly, and these bad behaviors will decrease relatively, so, in the later communication stage of the entire network, all the three trust factors will increase with the increased rounds of communication.

In order to verify the changes of three trust factors, we got Figures 7, 8, and 9. In Figure 7, the horizontal axis stands

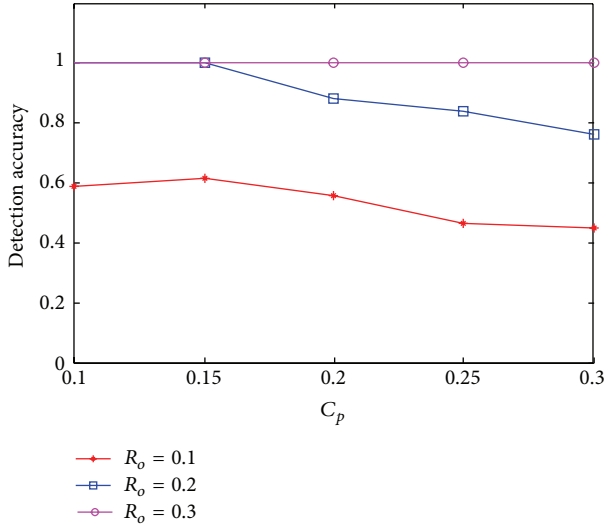


FIGURE 5: Proportion of malicious nodes and detection accuracy.

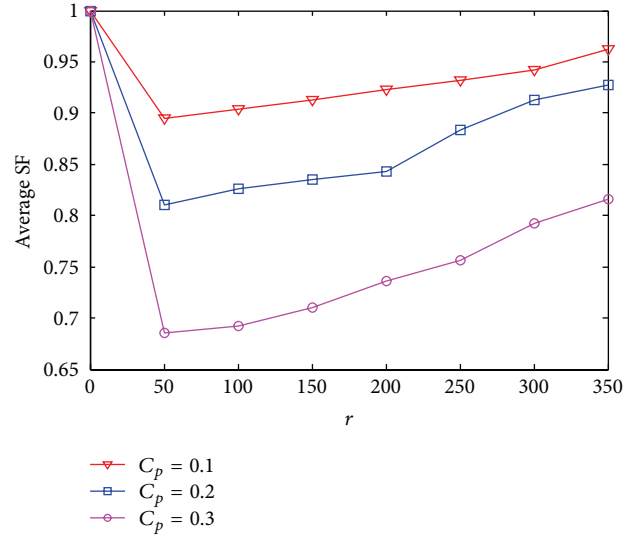


FIGURE 7: The average sending ratio.

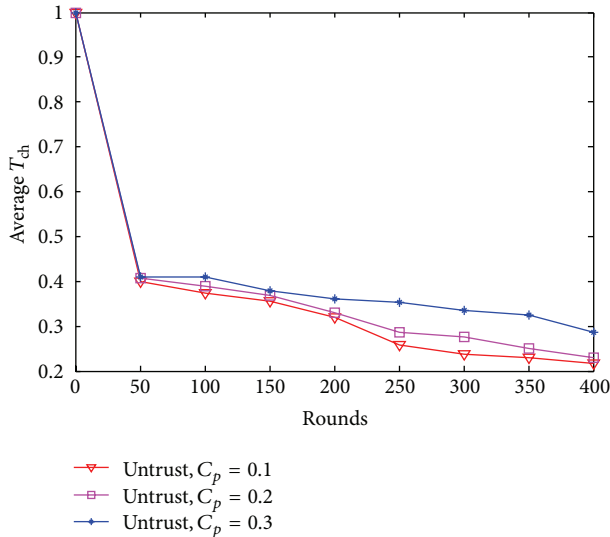


FIGURE 6: The average value of malicious node's trust.

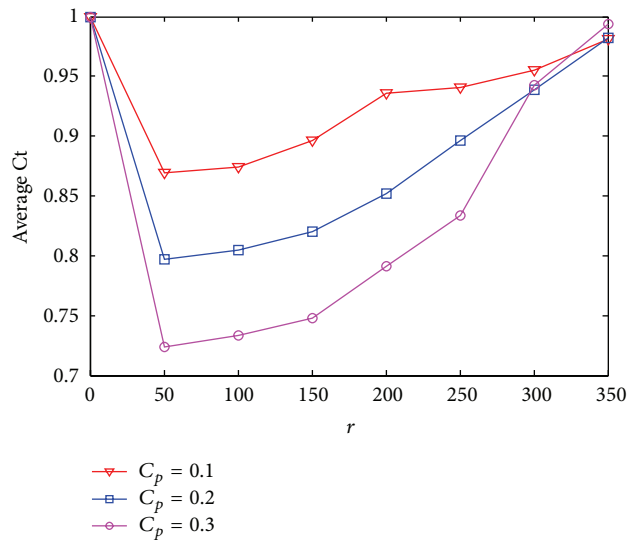


FIGURE 8: The average consistency ratio.

for number of communication round, and the vertical axis represents the average sending ratio. In Figure 7, no matter how much the proportion of malicious nodes is, the change of average sending ratio in the network suffers degradation in the first stage; with the increased number of communication, it will increase significantly. The greater the proportion of malicious nodes is, the faster the rate of decline is in the down phase. The horizontal axis and vertical axis represent number of communication round and the average consistency ratio respectively, in Figure 8. In Figure 9, the horizontal axis is number of communication rounds, and the vertical axis represents the average packet delivery ratio. The change of consistency ratio and average packet delivery is the same as that of the change of average sending ratio; they all suffer a degradation in the first stage and then increase significantly, no matter how much the proportion of malicious nodes is.

And they also have the characteristic; namely, the greater the proportion of malicious nodes is, the faster the rate of decline is in the down phase.

Compared with Figures 7, 8, and 9, We can see that the change of values of trust factors decreases first and then increases with the increase of the number of communication rounds. Sending factor's change is quite gentle, and the change of consistency factor and packet loss rate factor is relatively larger. This is because that we let malicious nodes send one more packet or one less packet than normal node in each round, in this experiment. And then statistics each node's sending rate after 50 rounds. The change of nodes' sending rate is not very high, so change of sending factor is slow.

The last part experiments the energy consumption in comparison with LEACH, LEACH-MF, CMRA, and TLES.

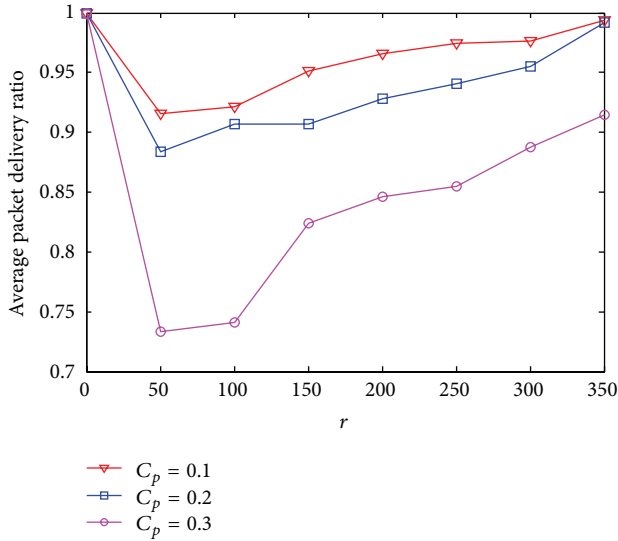


FIGURE 9: Average packet delivery ratio.

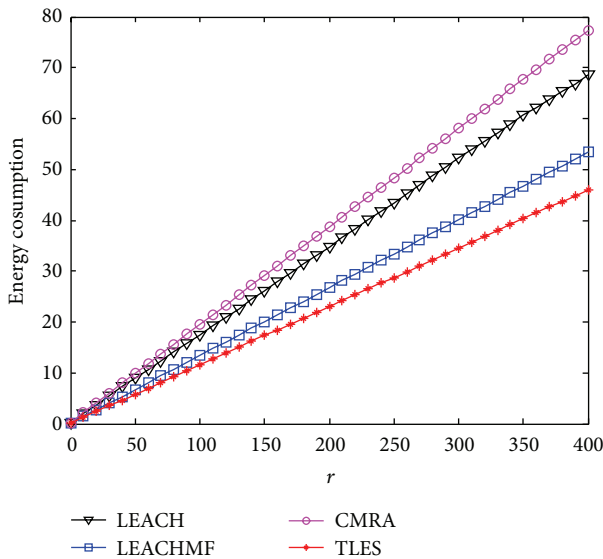


FIGURE 10: The comparison of energy consumption.

LEACH, LEACH_MF [17], and CMRA [18] are clustering routing protocols in WSN. Under the initial environment, we get the contrast Figure 10 of energy consumption of the four different algorithms. In Figure 10, the horizontal axis shows round number, the vertical axis represents the sum of all the network nodes' energy consumption, and the unit is J . As can be seen from Figure 10, TLES energy consumption is obviously smaller than the other three.

The number of rounds represents the lifetime of network in this simulation. The lifetime of network contains three kinds of definitions: the first node dies, half of nodes die, and the last node dies. In this experiment, we adopt the first definition (the first node dies) to count lifetime of network.

The comparison of energy consumption (LEACH, LEACH-MF, CMRA, and TLES) in the different scale of

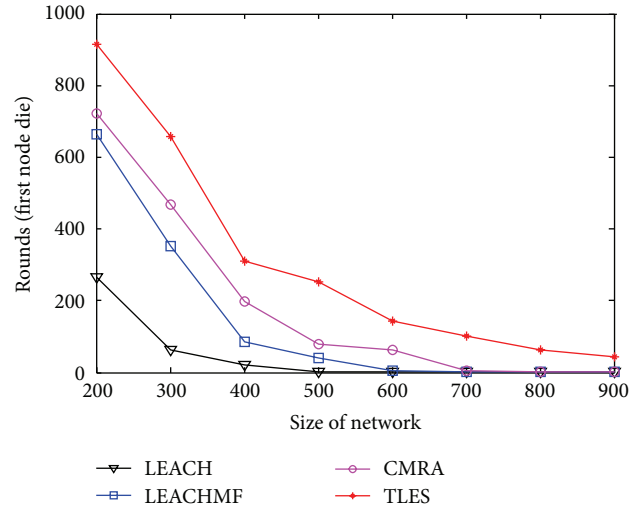


FIGURE 11: The relationship between the monitoring area and lifetime of network (the first node dies).

network is shown in Figure 11. Monitoring area is 200×200 , 300×300 , 400×400 , 500×500 , 600×600 , 700×700 , and 800×800 , respectively. Initial energy is $0.6 J$. In Figure 11, the horizontal axis represents the network scale; the vertical axis represents the lifetime (first node dies). With the increase of network size, most nodes' distance to the BS will increase. According to the communication model, we know that the increase of distance is bound to bring the increase of energy consumption. From Figure 11, we can see that the lifetimes of LEACH, LEACH-MF, CMRA, and TLES all decline with the increase of network scale. Although TLES is falling with the increase of scale of network, its survival time is longer than other algorithms under every scale.

Through Figures 10 and 11, we can know that TLES compared with LEACH, LEACH-MF, and CMRA has less energy consumption in each communication round and, with the increase of network scale, has the longest lifetime in the network.

6. Summary and Outlook

This paper proposed a secure topology protocol of WSN, that is, TLES. The trust mechanism used in TLES is introduced. Trust factors were defined by the node's historical behavior, and the trust value of each node was calculated according to the comprehensive value of direct trust and indirect trust, which are related to the trust factors. TLES uses the idea of clustering. First of all, the cluster heads were selected according to the trust value, residual energy, and density of nodes. Then, the cluster heads choose the next-hop node by the residual energy, the distance to BS, and degree of candidate node. After that, the construction of the whole network topology was built. Experimental results show that TLES can eliminate the malicious nodes in network effectively, so as to ensure the safety and rationality of node communication. At the same time, it can also reduce the energy consumption of the network.

The existing problems of this paper are focused on the following two aspects. First, this paper improves the average packet delivery ratio and increases the calculation leading to the increase of packet delay. Second, the mobile sensor network and heterogeneous network would become the new characteristics of network. It is important to figure out how to make improvement in the future.

Conflict of Interests

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

Acknowledgments

This work was sponsored by the Natural Science Foundation of Hunan Province, China (13JJ3091 and 14JJ3062), National Nature Science Foundation, China (61202462 and 61300036), and the Fundamental Research Funds for the Central Universities, China.

References

- [1] T. Eswari and V. Vanitha, "A novel rule based intrusion detection framework for Wireless Sensor Networks," in *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES '13)*, pp. 1019–1022, IEEE, Chennai, India, February 2013.
- [2] F. Gao, H.-L. Wen, L.-F. Zhao, and Y. Chen, "Design and optimization of a cross-layer routing protocol for multi-hop wireless sensor networks," in *Proceedings of the International Conference on Sensor Network Security Technology and Privacy Communication System (SNS & PCS '13)*, pp. 5–8, Nangang, China, May 2013.
- [3] Z. Li, N. Wang, A. Franzen et al., "Practical deployment of an in-field soil property wireless sensor network," *Computer Standards and Interfaces*, vol. 36, no. 2, pp. 278–287, 2014.
- [4] L. Yu, N. Wang, and X. Meng, "Real-time forest fire detection with wireless sensor networks," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WCNM '05)*, pp. 1214–1217, September 2005.
- [5] X. Liu, H. Zhao, X. Yang, and X. Li, "SinkTrail: a proactive data reporting protocol for wireless sensor networks," *IEEE Transactions on Computers*, vol. 62, no. 1, pp. 151–162, 2013.
- [6] J. Q. Zhang, R. Shankaran, M. A. Orgun et al., "A dynamic trust establishment and management framework for wireless sensor networks," in *Proceedings of the 8th International Conference on Embedded and Ubiquitous Computing (EUC '10)*, pp. 11–13, 2010.
- [7] J. Q. Duan, D. Y. Gao, D. Yang et al., "An energy-aware trust derivation scheme with game theoretic approach in wireless sensor networks for IoT applications," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 58–69, 2014.
- [8] J. Duan, D. Yang, H. Zhu, S. Zhang, and J. Zhao, "TSRF: a trust-aware secure routing framework in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 209436, 14 pages, 2014.
- [9] G. Zhan, W. Shi, and J. Deng, "Design and implementation of TARF: a trust-aware routing framework for WSNs," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 2, pp. 184–197, 2012.
- [10] R. A. Rajee and A. V. Sakhare, "Routing in wireless sensor network using fuzzy based trust model," in *Proceedings of the 4th International Conference on Communication Systems and Network Technologies (CSNT '14)*, pp. 7–9, 2014.
- [11] G. V. Crosby, N. Pissinou, and J. Gadze, "A framework for trust-based cluster head election in wireless sensor network," in *Proceedings of the 2nd IEEE Workshop on Dependability and Security in Sensor Network and Systems (DSSNS '10)*, pp. 24–28, 2006.
- [12] H. Safa, H. Artail, and D. Tabet, "A cluster-based trust-aware routing protocol for mobile ad hoc networks," *Wireless Networks*, vol. 16, no. 4, pp. 969–984, 2010.
- [13] W. R. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd International Conference on System Sciences*, pp. 1–10, Maui, Hawaii, USA, 2000.
- [14] W. Wang, F. Du, and Q. Xu, "An improvement of LEACH routing protocol based on trust for wireless sensor networks," in *Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '09)*, pp. 1–4, Beijing, China, September 2009.
- [15] A. Rezgoui and M. Eltoweissy, "TARP: a trust-aware routing protocol for sensor-actuator networks," in *Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS '07)*, pp. 1–9, IEEE, Pisa, Italy, October 2007.
- [16] 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.
- [17] H. Liu, Y. Fang, X. Hao, J. Dou, H. Li, and W. Bi, "Research of inter-cluster multi-hop routing algorithm for wireless sensor networks," in *Proceedings of the 3rd International Conference on Intelligent System and Knowledge Engineering (ISKE '08)*, vol. 1, pp. 1367–1372, Xiamen, China, November 2008.
- [18] 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.

Research Article

Sequence Alignment with Dynamic Divisor Generation for Keystroke Dynamics Based User Authentication

Jiacang Ho¹ and Dae-Ki Kang²

¹Department of Ubiquitous IT, Graduate School, Dongseo University, 47 Jurye-Ro, Sasang-Gu, Busan 617-716, Republic of Korea

²Department of Computer & Information Engineering, Dongseo University, 47 Jurye-Ro, Sasang-Gu, Busan 617-716, Republic of Korea

Correspondence should be addressed to Dae-Ki Kang; dkkang@dongseo.ac.kr

Received 28 November 2014; Revised 5 January 2015; Accepted 20 January 2015

Academic Editor: Fei Yu

Copyright © 2015 J. Ho and D.-K. Kang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Keystroke dynamics based authentication is one of the prevention mechanisms used to protect one's account from criminals' illegal access. In this authentication mechanism, keystroke dynamics are used to capture patterns in a user typing behavior. Sequence alignment is shown to be one of effective algorithms for keystroke dynamics based authentication, by comparing the sequences of keystroke data to detect imposter's anomalous sequences. In previous research, static divisor has been used for sequence generation from the keystroke data, which is a number used to divide a time difference of keystroke data into an equal-length subinterval. After the division, the subintervals are mapped to alphabet letters to form sequences. One major drawback of this static divisor is that the amount of data for this subinterval generation is often insufficient, which leads to premature termination of subinterval generation and consequently causes inaccurate sequence alignment. To alleviate this problem, we introduce sequence alignment of dynamic divisor (SADD) in this paper. In SADD, we use mean of Horner's rule technique to generate dynamic divisors and apply them to produce the subintervals with different length. The comparative experimental results with SADD and other existing algorithms indicate that SADD is usually comparable to and often outperforms other existing algorithms.

1. Introduction

In an era which is full of electronic services, people want to have more convenient and faster ways to assist their needs. This includes reading emails, searching information through online communication, transferring files, and paying bills online. For example, online file transfer involves storage mechanisms in a cloud system. To use these services for our private needs, we must register a login ID and password. As for the online bill payment, we must login with our ID and password before we can pay or transfer money to other users. However, it can happen that criminals detect our login ID and password and utilize our credentials to commit crimes such as stealing important files or money. Therefore, stronger and more secure authentication mechanism has to be designed and implemented to prevent these issues.

Considerable amount of authentication mechanisms has been introduced. One of the examples is a biometric system [1]. Biometric system can be divided into two folds,

which are physiology-based approach and behavior-based approach. Physiology-based approaches in the authentication system include use of iris, voice, and fingerprint. In contrast, behavior-based approach includes keystroke dynamics on keyboard, mouse, or smart phone. In this paper, we focus more on the behavior-based approach. Behavior-based approach has advantages that it is inexpensive, is easy to implement, and needs no extra hardware to operate. One possible drawback of the behavior-based approach is that its performance can be lower than that of physiology-based approach, but it can reduce the personal risk of a user. For example, in fingerprint-based authentication, an imposter can cut off the genuine user's hand in order to access the security system. However, this kind of attack is ineligible for behavior-based approach. If the imposter wants to break a system, his choice can be limited in either forcing the genuine user to type it or emulating the genuine user's typing by practice. Therefore, there is still a considerable amount of researchers performing behavior-based research because

they believe that keystroke dynamics can improve the security and it will be a common approach to be used in the future [2–10].

Keystroke dynamics concerns timing details of people's typing data [11]. The timing detail means the duration between pressing a key and releasing a key or vice versa. Keystroke dynamics can be applied with the machine learning to discover knowledge about people's typing behavior [8], emotion [12], gender [3], dominant hand [4], and so forth. In this paper, we associate the typing behavior with the authentication system. It is reasonable to use the keystroke dynamics in the authentication system because some users have special typing patterns. For example, some people use only one hand to type whereas the others use both hands [4]. Moreover, Syed et al. [8] have proven three interesting hypotheses in their research. One of the evidence is that users present significantly dissimilar pattern when typing. Another hypothesis is about the relationship between users' typing ability and the event sequence. The event sequence is defined as the sequence of key-up and key-down events with actual key values incorporated. More explanation for event sequence can be reviewed in their paper [8]. In their result, it is shown that there is no correlation between users' typing skill and the event sequence. Last hypothesis discusses the effect of habituation on the event sequences of a user. It reveals that the keystroke dynamics data that is typed later is more representative than the data which is typed earlier. This means that it is difficult for imposters to imitate legitimate users' typing patterns. Besides typing behavior, we can observe that different people have different walking styles (or gait). That is, we can conjecture a person correctly just by hearing the footsteps without seeing him/her. It can be possible that, sometimes, we can surmise a person correctly by looking at the appearances and walking style from behind. In summary, we can identify a person easily by using these special characteristics or so-called special habits. These characteristics or habits include typing behavior.

The common timing details that we can obtain from keystroke dynamics are dwell time and flight time. Dwell time, also known as duration time [9], is the length of time between when a key is pressed and when a key is released. Flight time, also known as interval time [9], however, is the length of time between when a key is released and when a next key is pressed. More details are discussed in Section 4.

There is a considerable amount of machine learning algorithms introduced for keystroke dynamics such as naïve Bayes [13], support vector machines (SVM) [10], nearest-neighbor [2], and Euclidean distance measure [14]. In this paper, we choose sequence alignment algorithm. Sequence alignment algorithm is fundamentally concerned about measuring the similarity between multiple sequences. Suppose a user logs in an account in any platform, the system has to check the existence of her identification information before it checks her password. It is well known that this password matching involves complicated encryption of plain text in symmetric password systems [15] or modular arithmetic operations for asymmetric password systems [16]. However, in a more abstract view, this is basically string matching. When a system checks the password, firstly it checks the first

letter from the currently inserted password with the first letter from the stored password in the database, followed by the second letter, the third letter, and so forth until all letters are verified. Sequence alignment is a more general and stronger algorithm which measures the similarity between objects. Hence, we consider that sequence alignment algorithm is an appropriate algorithm to be used in this paper.

Of the current research we are aware of, there is still no specific algorithm to be used as the common algorithm in keystroke dynamics research. However, in Revett's research [7], he shows that the sequence alignment algorithm has performed sufficiently when it is applied into the keystroke dynamics. Besides that, some researchers have provided new idea into keystroke dynamics. Giot and Rosenberger's [3] research introduces a new soft biometric for keystroke dynamics based on gender recognition. Idrus et al. [4] also introduce more valuable information such as the type of hand used, age, and the dominant hand. These extra amounts of information can be used as reference in order to help to improve the performance of algorithms when the algorithms are applied into keystroke dynamics. Furthermore, Syed et al. [8] show the concept of event sequences used in the keystroke dynamics. This event sequences help to distinguish the typing behavior of a user.

The next section describes sequence alignment algorithm. Section 3 discusses the proposed method. Sections 4 and 5 explain the experimental method and the results of the experiment, respectively. The final section presents several concluding remarks and future research issues.

2. Sequence Alignment Algorithm

Sequence alignment is an algorithm that calculates the similarity among two or more sequences [17]. This algorithm is widely used in bioinformatics areas such as deoxyribonucleic acid (DNA) sequences, ribonucleic acid (RNA) sequences, or protein sequences. In Revett's [7] research, he has applied the sequence alignment algorithm into the keystroke dynamics and obtained encouraging results. In this paper, we show the performance comparison of our proposed method, Revett's sequence alignment algorithm, and other previous work.

The keystroke dynamics is generated in timestamp format (millisecond). Since values in timestamp format vary and can be infinity, it is inappropriate to apply keystroke dynamics to sequence alignment algorithm directly. Therefore, we have to discretize the timestamp into subintervals. Each subinterval will represent a different category. For example, this process is similar to a questionnaire construction. We usually allow a user to choose few options, such as "strongly disagree," "disagree," "neither disagree nor agree," "agree," and "strongly agree." Sometimes, we also just make it shorter to three options which are "disagree," "neither disagree nor agree," and "agree." But usually we just put maximum options to five or six options. We do not put too many categories into the questionnaire because it is hard to be analyzed later. Revett [7] has used twenty categories (i.e., twenty bins) in his research. These twenty categories are extracted from the letters of amino acid. These letters represent "ACDEFGHIKLMNPQRSTVWY." With the use of twenty bins in

keystroke dynamics, it becomes much suitable for sequence alignment algorithm to be used.

We explain the algorithm design of sequence alignment in the following paragraphs. Firstly, we have to get the difference of the time interval from a feature, for instance, dwell time. This time interval difference is obtained from the difference between maximum time and minimum time. The maximum time of dwell time means the longest time for a user to press a key and release a key. The minimum time of dwell time, on the other hand, is the shortest time for a user to press a key and release a key. The formula is defined by

$$\text{diff}_j = \max_j - \min_j, \quad (1)$$

where j is the number of the column (attribute).

After we obtain the difference of the time interval from an attribute, we have to divide the difference of time intervals into twenty subintervals. The length of subintervals is defined by

$$\text{rangeDiff}_j = \frac{\text{diff}_j}{20}, \quad (2)$$

where j is the number of the column. The reason why we have to get the rangeDiff is because we need to know the length of each category. If a new dwell time is close to the minimum time of dwell time, it will be categorized as letter A. The next category is C, followed by D, and so forth. If a new dwell time is close to the maximum dwell time, then it will be categorized as Y. At the end of calculation, each category has the exact same length, rangeDiff, because static divisor is used. Labeling (mapping) a new time can be formulated by

$$\text{label}_{i,j} = \left\lceil \frac{\text{new_time}_{i,j} - \min_j}{\text{rangeDiff}_j} \right\rceil, \quad (3)$$

where i is the number of the row (also known as entry) and j is the number of the column. If a new time of a feature is less than the minimum time of the feature or greater than the maximum time of the feature (it mostly happens in the testing phase), it will be categorized as a different alphabet letter and is excluded in the alignment. Equations (1), (2), and (3) are the equations used to map the keystrokes to proper alphabet letters.

Once a row of data (entry) is changed to the corresponding alphabet letters, we run the sequence alignment algorithm. One point is scored if the label is matched in an attribute. Otherwise, no points are scored. The score is described as

$$\text{Score}_j \begin{cases} 1, & \text{if it is matched,} \\ 0, & \text{if it is not matched,} \end{cases} \quad (4)$$

where j is the number of the column. When an entry is completely verified, we sum all the scores. It is then defined by

$$\text{Final_Score}_i = \sum_{j=1}^N \text{Score}_j, \quad (5)$$

where i is the number of the row, j is the number of the column, and N is the total number of the columns. This final score is then compared with the user-specified threshold. The higher the final score is, the higher the possibility that the new data is recognized as the genuine user is. In order to present a clear description for the whole algorithm design, we display the pseudocode of this algorithm in Algorithm 1.

In Algorithm 1, we can see that there are training phase and testing phase. In the training phase, the algorithm starts by calculating the maximum and the minimum for each attribute. It calculates the range difference (rangeDiff) for each attribute. Finally, it calculates labels to construct models.

In the testing phase, the algorithm converts the given testing data to alphabet letter format based on the minimum point obtained at step 3 and the range difference from step 7 in Algorithm 1. After that, it executes the conversion loop described from steps 9 to 14 for once on the test data. This is the conversion process described in step 1 in the testing phase.

After the conversion, for the actual recognition, for each row in the model, the algorithm calculates the score (Final_Score) for the row and the test data. This score is a summation of match scores (score in Algorithm 1) between the attribute in the row of the model and the attribute in the test data. Finally, the statistical summary of information including maximum, minimum, median, mean, and mode of the scores is calculated for the comparison with corresponding thresholds.

3. Sequence Alignment with Dynamic Divisor Generation Algorithm

In this paper, we propose sequence alignment with dynamic divisor generation (SADD) algorithm. SADD checks the degree of sufficiency of the dataset and then provides a proper divisor instead of static divisor as shown in (2) for each attribute. We show the steps to check the degree of sufficiency of the dataset and the reason we used dynamic divisor instead of static divisor in next paragraphs.

As a human, we are unfamiliar with a new thing immediately from the beginning. We have to practice a few times to get accustomed to the new thing. For example, consider an athlete who wants to run a 100-meter track in 10 seconds. However, this is nearly impossible if she is a beginner. She has to train hard and practice regularly. The time record from the first day until the day she manages to run a 100-meter track in 10 seconds could be illustrated as a graph which is shown in Figure 1. It is worth noting that, after few months of training, she will find it very difficult to reduce the time (i.e., less than 10 seconds) to finish the 100-meter track. This is because there is a limitation point of where we can reach even despite how hard we have trained and practiced. We called this phenomenon "realm point." Realm point refers to the temporal or spatial point when the user is accustomed to something or the user has a habit on something. This is also a reason why the world record of 100 meters currently remained at 9.58 seconds.

The phenomenon (i.e., realm point) that we have discussed previously can be applied for most activities including

Input: Training data extracted from a genuine user, A . Testing data extracted from a genuine user or an imposter, B .

In the training phase:

- (1) **for** each attribute j **do**
- (2) $\max[j] \leftarrow \max(\max[j], A_j)$
- (3) $\min[j] \leftarrow \min(\min[j], A_j)$
- (4) **end for**
- (5) **for** each attribute j **do**
- (6) $\text{diff} \leftarrow \max[j] - \min[j]$
- (7) $\text{rangeDiff}[j] \leftarrow \text{diff}/20$
- (8) **end for**
- (9) **for** each row i and each attribute j in A **do**
- (10) $\text{label} \leftarrow \text{ceil}((A_{i,j} - \min[j])/\text{rangeDiff}[j])$
- (11) **if** $\text{label} = 0$
- (12) $\text{label} \leftarrow 1$ // If the data is exactly the same value as minimum time, then it should categorize as label A
- (13) $\text{model}[i, j] \leftarrow \text{label}$ // We just keep it as 1, 2, 3, ..., 20 to represent A, C, D, ..., Y
- (14) **end for**

In the testing phase:

- (1) Convert the B to alphabet letter format based on the minimum point and the range difference from step 3 and step 7 respectively, from training phase, and then we just run one time from step 9 to step 14 from training phase.
- (2) **for** each row i in model **do**
- (3) **for** each attribute j in B **do**
- (4) **if** $B[j] = \text{model}[i, j]$
- (5) $\text{Score}[j] \leftarrow 1$
- (6) **end for**
- (7) $\text{Final_Score}[i] \leftarrow \text{sum}(\text{Score})$
- (8) **end for**
- (9) $\text{Checking_Score} \leftarrow \text{mean}(\text{Final_Score})$ // Can be max, min, median, mean, and mode
- (10) **return** Checking_Score

Output: The score then used to compare the threshold

ALGORITHM 1: The pseudocode to explain the sequence alignment algorithm.

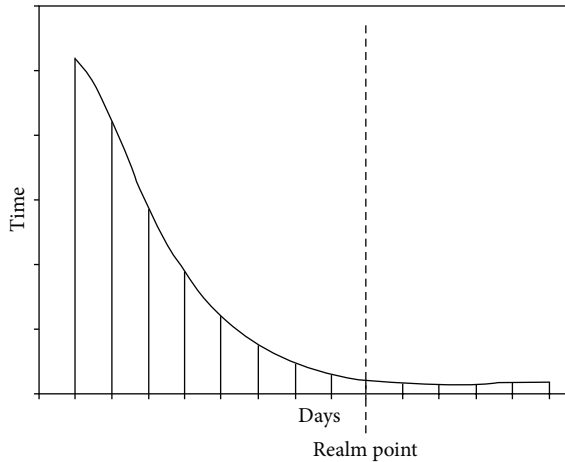


FIGURE 1: The time of practicing a new action versus the day of practicing it.

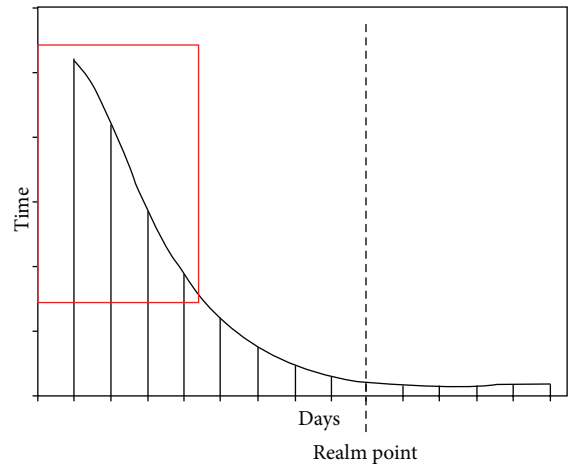


FIGURE 2: The worst case of the data used in the experiment or in real authentication system.

the typing speed of a password. Unfortunately, it is difficult to know how many hours, days, weeks, months, or years needed to reach the realm point of typing speed. Every user will have different time to reach realm point even with all the other conditions fixed. We do not know if the users are reaching the realm point or not in the beginning of the experiment,

and real authentication systems do not know this either. For the best case, the data we collected cover from the beginning of the day to the realm point (or after realm point). For the worst case, it can be from the beginning of the day until the middle of the days, as shown in Figure 2.

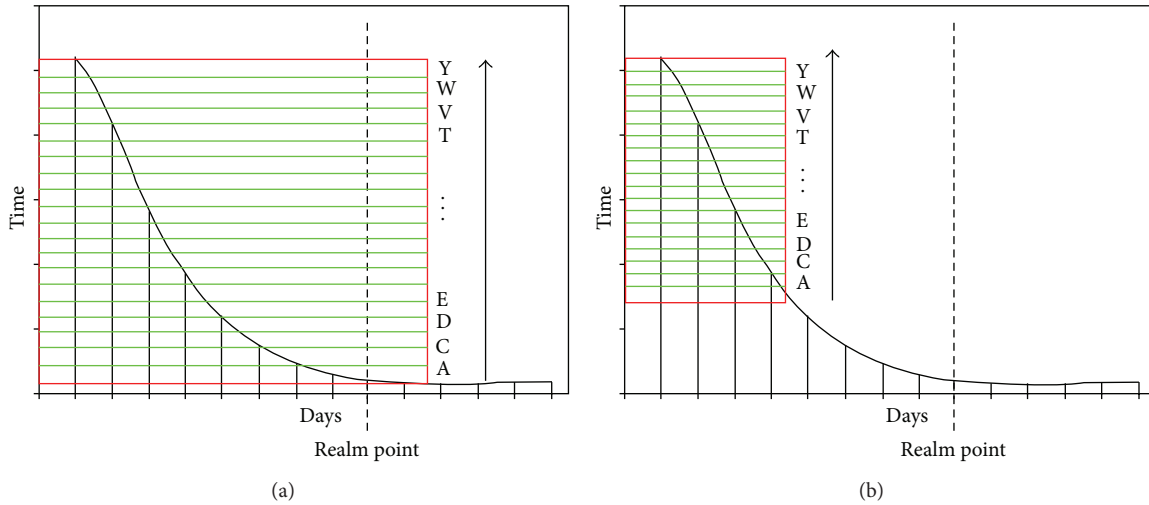


FIGURE 3: (a) The best case after the data is converted into a sequence. (b) The worst case after the data is converted into a sequence.

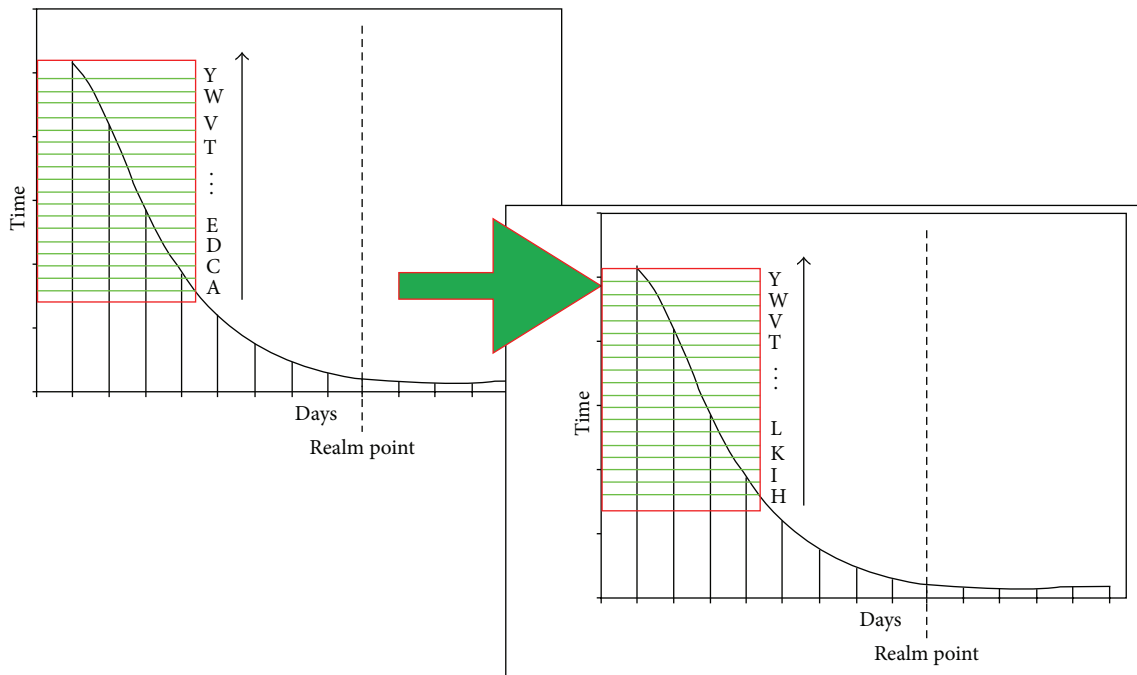


FIGURE 4: Data mapping with the amino acid letters in the worst case. Note that mapping is started from the middle of the letters.

For (1) and (2), we have to find each category in each subinterval. Figures 3(a) and 3(b) present the graph when the best case and worst case are applied with (1) and (2). Note that the intervals highlighted inside the red rectangle box are the ones mapped to the alphabet letters and are stored as a model in the database. This constructed model will be used to verify the new data. In particular, for the worst case, it can be seen that the categories generated do not cover whole information. Hence, if the genuine user gets accustomed to the password typing, then the minimum time of the new data she has inserted is lower than the minimum time in the database. Thus, in turn, this new data has high probability to be rejected as the imposter because it is out of category. In

order to avoid this problem, we propose SADD in this paper. We use a subset of alphabet letters (e.g., 15 bins) and apply it to the dataset if its contents are insufficient. The idea is illustrated in Figure 4. We point out this problem to ensure that the model that we used in the database is always faultless and still can be used as a rewind case like when a user is sick and her typing speed becomes slower than usual but her typing behavior is still the same.

Now, we explain the proposed algorithm design. Firstly, (1) remains the same. The main procedure of SADD is to find the correct and suitable divisor used in (2). To find a proper divisor in each attribute, we exploit Horner's rule [18] to get the mean. Figure 5 shows the illustration of the way we use

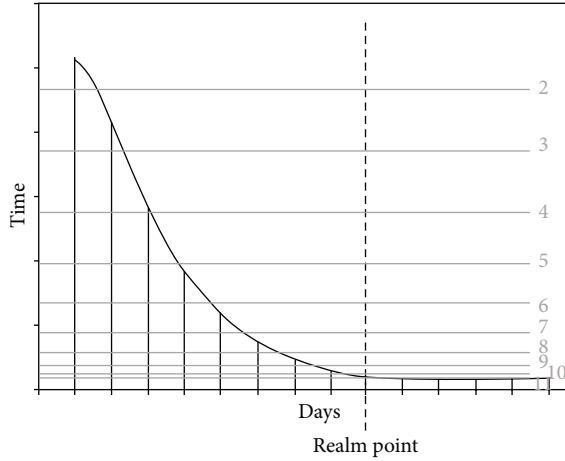


FIGURE 5: Calculation of mean with Horner's rule.

Horner's rule to calculate the mean. The grey line stated with 2 (known as line 2) on the right hand side is the mean of the first point (maximum point) and the second point. Line 3 is the mean of line 2 and third point. Line 4, however, is the mean of line 3 and the fourth point. The calculation is repeated by summing the previous output and the new data and then halving the sum. The repetition is terminated if it is reached to the end of the data. As shown in Figure 5, line 11 is the mean of line 10 and eleventh point. We also can observe that line 11 is closed to the realm point. By obtaining line 11, we can conclude that this user has reached to the realm point. The calculation for mean of each attribute by using Horner's rule is defined by

$$\text{mean}_j = \frac{((((((x_1 + x_2) / 2) + x_3) / 2) + \dots) / \dots) + x_i}{2}, \quad (6)$$

where i is the number of the row and j is the number of the column.

From line 11 in Figure 5, we observe that the calculated mean is near to the minimum point (i.e., realm point) and is far from the maximum point (i.e., beginning point). To get the proper divisor, we have to find the ratio of the mean from minimum point and the maximum point to the mean. The formula is described as

$$\text{ratio}_j = \frac{(\text{mean_HR}_j - \min_j)}{(\max_j - \min_j)}, \quad (7)$$

where j is the number of the column and $0.5 \leq \text{ratio}_j < 1$.

Once we obtain the ratio, we calculate the divisor by

$$\text{divisor}_j = I - (I * \text{ratio}_j), \quad (8)$$

where j is the number of the column, I is 20 for twenty bins, and $10 \leq \text{divisor}_j \leq 20$.

After that, we replace the twenty values from (2) to this new divisor. It is calculated as

$$\text{rangeDiff}_j = \frac{\text{diff}_j}{\text{divisor}_j}, \quad (9)$$

where j is the number of the column.

Consider the worst case shown in Figure 3(b). Note that the first letter should not be labeled as "A" but a certain letter in the middle of the alphabet. Therefore, we modify (3) as follows:

$$\text{label}_{i,j} = \left\lceil \frac{\text{new_time}_{i,j} - \min_j}{\text{rangeDiff}_j} \right\rceil + (I - \text{divisor}_j), \quad (10)$$

where i is the number of the row, j is the number of the column, and I is for the twenty bins. This means that the new data which is closer to the maximum point will be classified as "Y" and followed by "W," "V," "T," and so forth. The calculation of score and the final score remains the same as (4) and (5). In order to clearly explain our proposed algorithm, we provide Algorithm 2.

In the training phase of Algorithm 2, we calculate the mean by Horner's rule in steps 5 to 9. We calculate ratio in (7) at step 11, divisor in (8) at step 12, and range difference in (9) at step 13. Label assignment described in (10) is implemented at step 16 of Algorithm 2.

4. Experimental Method

4.1. Training and Testing Phase. In the keystroke dynamics, there are six common timing elements that we can use between two different keystrokes. They are as follows.

- (i) Hold (H): it is a duration time (or a dwell time) of pressing a key:
 - (a) a holding time of the first key, H1;
 - (b) a holding time of the second key, H2.
- (ii) Up-down (UD): it is a duration time (or a flight time) between key-up of the first key and key-down of the second key.
- (iii) Down-down (DD): it is a duration time between key-down of the first key and key-down of the second key; it is the sum of H1 and UD.
- (iv) Up-up (UU): it is a duration time between key-up of the first key and key-up of the second key; it is the sum of UD and H2.
- (v) Down-up (DU): it is a duration time between key-down of the first key and key-up of the second key; it is the sum of DD and H2.

In our experiment, we use the CMU benchmark dataset [5]. This dataset consists of four elements, which are H1, H2, UD, and DD. However, we doubt that the DD element is not an appropriate element to be used in sequence alignment or similar algorithms. This is because sequence alignment algorithm is comparing attribute by attribute. In this case,

Input: Training data extracted from a genuine user, A . Testing data extracted from a genuine user or an imposter, B .

In the training phase:

- (1) **for** each attribute j **do**
- (2) $\max[j] \leftarrow \max(\max[j], A_j)$
- (3) $\min[j] \leftarrow \min(\min[j], A_j)$
- (4) **end for**
- (5) **for** each attribute j **do**
- (6) $\text{mean} \leftarrow A_1$
- (7) **for** each row $i + 1$ **do**// Start from second row
- (8) $\text{mean} \leftarrow (\text{mean} + A_i)/2$
- (9) **end for**
- (10) $\text{diff} \leftarrow \max[j] - \min[j]$
- (11) $\text{ratio} \leftarrow (\text{mean} - \min[j])/\text{diff}$
- (12) $\text{divisor}[j] \leftarrow 20 - (20 * \text{ratio})$
- (13) $\text{rangeDiff}[j] \leftarrow (\text{diff}/\text{divisor}[j])$
- (14) **end for**
- (15) **for** each row i and each attribute j in A **do**
- (16) $\text{label} \leftarrow \text{ceil}((A_{i,j} - \min[j])/\text{rangeDiff}[j]) + (20 - \text{divisor}[j])$
- (17) **if** label = 0
- (18) $\text{label} \leftarrow 1$ // If the data is exactly the same value as the minimum time, then it should be categorized as label A
- (19) $\text{model}[i, j] \leftarrow \text{label}$ // We just preserve it as 1, 2, 3, ..., 20 to represent A, C, D, ..., Y
- (20) **end for**

In the testing phase:

- (1) Convert the B to the alphabet letter format based on the minimum point the range difference from step 3 and step 7 respectively, from training phase. And then we run from step 15 to step 20 for just once time.
- (2) **for** each row i in model **do**
- (3) **for** each attribute j in B **do**
- (4) **if** $B[j] = \text{model}[i, j]$
- (5) $\text{Score}[j] \leftarrow 1$
- (6) **end for**
- (7) $\text{Final_Score}[i] \leftarrow \text{sum}(\text{Score})$
- (8) **end for**
- (9) $\text{Checking_Score} \leftarrow \text{mean}(\text{Final_Score})$ // Can be max, min, median, mean, and mode
- (10) **return** Checking_Score

Output: The score then used to compare the threshold

ALGORITHM 2: The pseudocode of sequence alignment with dynamic divisor generation (SADD) algorithm.

it is checking element by element. Since DD is the sum of H1 and UD, there is a possibility to obtain the same value of DD with different value of H1 and UD. For instance, consider the following example with two different instances as shown in Figure 7.

Assume that Instance number 1 is the data that we have collected from a genuine user and it is used as model and Instance number 2 is a new data inserted by an imposter. Since it is a short example, we omit the procedure to convert the timestamp format into consensus sequence (label format). As we explain above, the sequence alignment algorithm is checking element by element. The first element it checks is H1. Based on (4), since the H1 of Instance number 2 is mismatched with the H1 of Instance number 1, the zero score is given. Next, the algorithm checks the UD element. They are mismatched too, and so zero score is given. Finally, the algorithm checks the last element, DD. Since they are matched to each other, one score is given. If the threshold given from a system is one, then Instance number 2 will be predicted as a genuine user. Otherwise, it will be rejected as an imposter's. If the system predicts Instance number 2 as

a genuine user, then it will cause a lot of loss to the genuine user. Besides that, the example we show is a short example. However, in the real life, there can be a considerable amount of DD elements. If it is too fortunate for an imposter whose entire DD elements are matched (due to DD element being the sum of H1 and UD) with some entries of genuine users in the stored model and the threshold given is the total number of DD elements, then the system will accept this instance as a genuine user. Then, the imposter can access the system.

Therefore, in order to discover the effectiveness of DD elements in the authentication, we create another dataset from benchmark dataset which is having no DD elements (by removing all DD attributes from original dataset). Other than that, we also want to evaluate how effective it is to use all elements in the authentication. Hence, we create another dataset from benchmark dataset which is having extra UU and DU elements (by adding UU and DU attributes into original dataset). Basically, the difference between these three datasets is the number of the attributes used in the experiment. The first dataset (Dataset number 1) consists of 31 attributes, the second dataset (Dataset number 2) consists

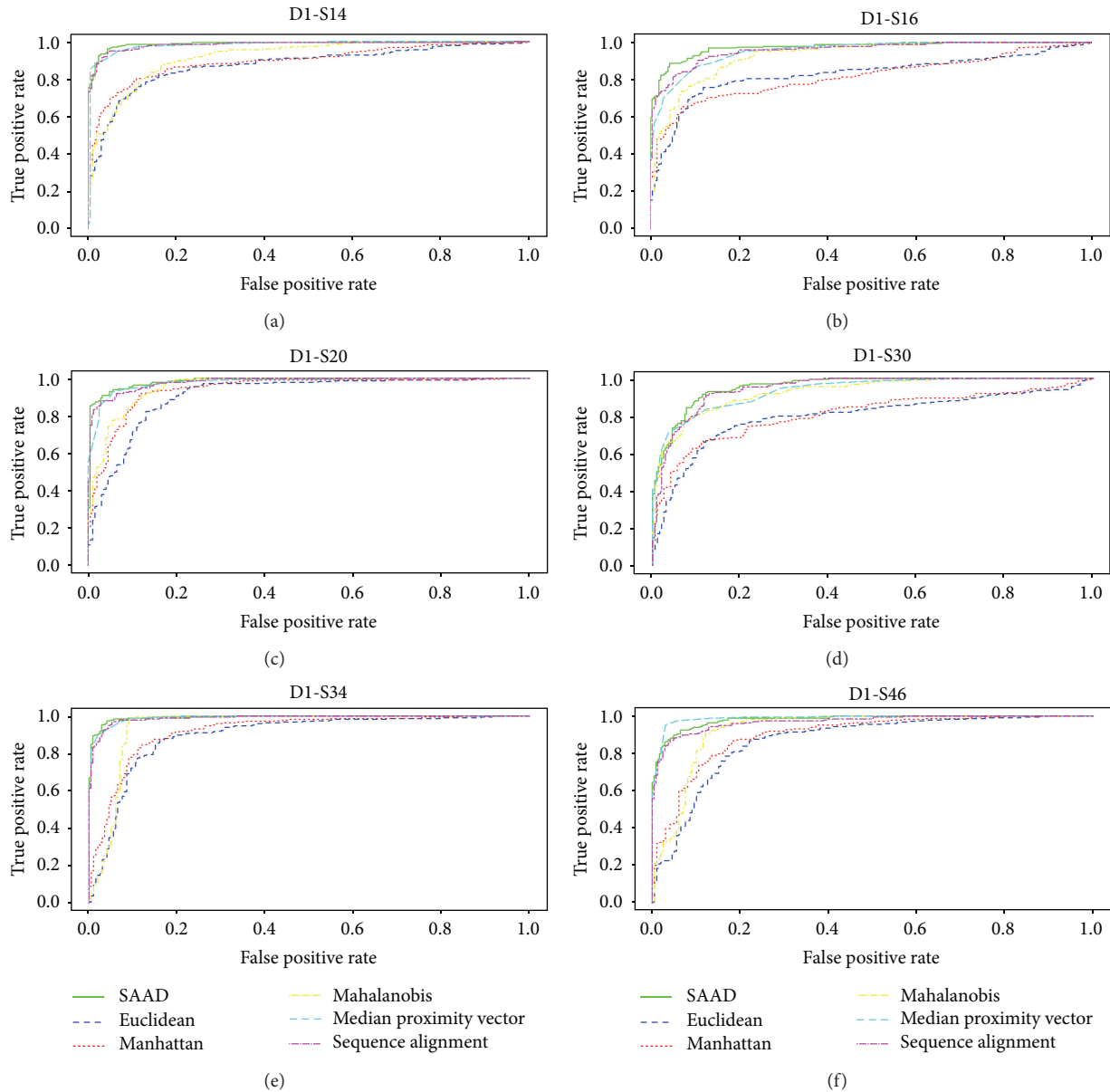


FIGURE 6: Examples of ROC curves from any 6 subjects in dataset number 1. Note that D stands for dataset and S stands for subject.

Instance#1: H1: 10 ms, UD: 14 ms, DD: 24 ms

Instance#2: H1: 12 ms, UD: 12 ms, DD: 24 ms

FIGURE 7: Instance number 1 is the data collected from a genuine user and stored as a model in database and Instance number 2 is the data collected from an imposter.

of 21 attributes, and the last dataset (Dataset number 3) has 51 attributes.

As for the benchmark dataset and two extra datasets we have created, each dataset contains 20,400 typing password entries. In each dataset, it consists of 51 subjects involved in this experiment. Each subject has 400 entries. In our experiment, during the training phase, we select one subject as genuine user and the remaining 50 subjects as imposters.

First 200 entries of the chosen subject are used as training data. However, the remaining 200 entries of the chosen subject are used as testing data. Besides that, we obtain the first five entries from the remaining 50 subjects as testing data too. In total, there are 450 entries used in the testing phase. Based on Killourhy and Maxion [5] research, the reason they use the top five from the remaining 50 subjects is because they assume the imposter is unfamiliar to the password. In our experiment, we will operate at least 51 times in each dataset, and thus the number of total runs for three datasets will be 153 runs.

4.2. Performance of Evaluation Using ROC Curves. The main performance of evaluation in our experiment is using receiver operating characteristic (ROC) curves. We compare our algorithm and other approaches by showing the graph in

TABLE 1: The average of equal error rates with the standard deviation of equal error rate (after plus-minus sign) for six algorithms in three datasets. The highest performance of the algorithm is in bold.

Algorithm	EER		
	Dataset number 1	Dataset number 2	Dataset number 3
SADD	0.076 ± 0.060	0.078 ± 0.060	0.082 ± 0.062
Sequence alignment	0.084 ± 0.061	0.079 ± 0.053	0.087 ± 0.061
Median vector proximity	0.080 ± 0.060	0.094 ± 0.069	0.081 ± 0.060
Mahalanobis	0.110 ± 0.065	0.110 ± 0.065	0.110 ± 0.065
Manhattan	0.153 ± 0.092	0.144 ± 0.090	0.154 ± 0.089
Euclidean	0.171 ± 0.095	0.170 ± 0.097	0.169 ± 0.093

ROC curves. One of the examples of ROC curves is shown in Figure 6. The label at y -axis is true positive rate. True positive rate is the rate when an imposter is detected. And the x -axis labels the false positive rate, which is the rate when a genuine user is detected as an imposter. We can calculate the equal error rate (EER) from the ROC curve. EER is the point that is intersected by diagonal line and the curve line. We can conclude that an algorithm's performance is higher than others if the curve line is closer to the 1.0 point y -axis or the area under the curve (AUC) is larger than other algorithms.

5. Experimental Results

As aforementioned, we run three different datasets in our experiment. The first dataset is the CMU benchmark dataset, the second dataset is without the DD element, and the last dataset is with extra UU and DU elements. Table 1 shows the performance of our algorithm (SADD), sequence alignment, median vector proximity [6], Mahalanobis distance, Manhattan distance, and Euclidean distance. Table 1 has shown that our proposed algorithm produced higher performance than other previous work. Although the results for Dataset number 3 are not the best among the six algorithms, our algorithm, SADD, shows only 0.001 difference, in terms of EER, compared with the median vector proximity algorithm. Based on the results, it can be seen that our algorithm is comparable or outperforms in any case, either with or without some elements (i.e., UU, DD, UD, and DU elements).

In order to observe the performance of each algorithm in each subject, we produce the area under curve (AUC) in Tables 2, 3, and 4 with Dataset number 1, Dataset number 2, and Dataset number 3, respectively. If the total of AUC is close to value 1, then it means that the classification of that particular algorithm is desirable. These results are related with those in Table 1. We can view from Tables 2 and 3 that SADD consists of highest value of AUC. Therefore, SADD has shown the highest performance with Dataset number 1 and Dataset number 2 in Table 1. However, Table 4 shows that median vector proximity has highest value of AUC, and this is the reason why median vector proximity has

highest performance with Dataset number 3 in Table 1. From Table 3, it is worthwhile to note that although median vector proximity shows overall higher accuracy, SADD has more wins (i.e., 20 wins) than median vector proximity has (i.e., 19 wins) for Dataset number 3.

Besides step 9 in the testing phase from Algorithms 1 and 2, we comment that we can use the statistical summary information such as minimum, maximum, mean, median, or mode when we calculate the checking score. In order to show the effectiveness of the statistical metric in the experiment, we provide the results with different statistical metrics in the form of the average of the EER with its standard deviation in Table 5. Interestingly, the result with mean metric has the higher performance than other statistical metrics such as median or mode. Furthermore, our proposed algorithm shows the highest performance in three datasets.

In addition, we operate an experiment with different number of the bins (different number of the categories) to test the effectiveness of the number of bins toward our proposed algorithm and the sequence alignment algorithm with static divisor. We show the result in Table 6. The statistical metric that we used to operate in this experiment is mean. Table 6 indicates that 20 bins or 30 bins have almost the same performance. However, depending on the dataset, we have to use different number of bins in order to produce optimal performance. As the results shown in Table 6, Dataset number 1 and Dataset number 2 have to use 20 bins to perform the highest results for SADD algorithm, but 30 bins have to be used in Dataset number 3 for SADD algorithm in order to obtain the highest results. Meanwhile, 30 bins are much appropriate to be used with SA algorithm in Dataset number 1 and Dataset number 3 and 20 bins, on the other hand, are much suitable for Dataset number 2.

6. Related Work

In Revett's research [7], he shows that the sequence alignment algorithm has performed effectively when it is applied into the keystroke dynamics. He also provides the steps of applying sequence alignment in the keystroke dynamics. However, there is still much more to improve as we have described in Section 3. Therefore, we have proposed our method named sequence alignment with dynamic divisor generation (SADD) algorithm which checks the degree of sufficiency of the dataset and provides a proper divisor to each attribute. Our result proves that SADD has higher performance than the sequence alignment algorithm using static divisor in every attribute.

Al-Jarrah [6] has proposed the median proximity vector in his paper. He provides a very good performance of this algorithm by using median instead of mean in his experiment. In our proposed algorithm, we apply various statistical metrics in the experiments to find the most appropriate measure.

Giot and Rosenberger's [3] research introduces a new soft biometric for keystroke dynamics based on gender recognition. Besides that, the similar study from Idrus et al. [4] also introduces more valuable information such as the number of hands used (i.e., one hand or both hands),

TABLE 2: The result of AUC for six algorithms in Dataset number 1. The highest value of AUC result is in bold.

Subject	Algorithm					
	SADD	Sequence alignment	Median vector proximity	Mahalanobis	Manhattan	Euclidean
1	0.91294	0.92024	0.91255	0.84844	0.81886	0.79240
2	0.97029	0.96491	0.96794	0.88018	0.87374	0.86002
3	0.98542	0.97690	0.95917	0.96926	0.89572	0.84620
4	0.99472	0.99249	0.99634	0.97216	0.96588	0.93968
5	0.97759	0.97771	0.97867	0.96218	0.93860	0.95596
6	0.98896	0.98872	0.98927	0.97972	0.94190	0.92976
7	0.99936	0.99835	0.99800	0.98944	0.99448	0.99088
8	0.99395	0.99359	0.99498	0.96762	0.95878	0.95658
9	0.99918	0.99846	0.99933	0.97772	0.97190	0.93394
10	0.99970	0.99978	0.99311	0.96102	0.96404	0.94450
11	0.98905	0.99086	0.96766	0.93818	0.87728	0.85674
12	0.89676	0.85888	0.87452	0.91010	0.81641	0.82154
13	0.99871	0.99626	0.99809	0.99184	0.98328	0.97100
14	0.98766	0.98363	0.98269	0.92130	0.89508	0.87682
15	0.99442	0.99460	0.99430	0.97628	0.95916	0.93830
16	0.97430	0.95875	0.95498	0.93130	0.81956	0.83250
17	0.96347	0.96203	0.96723	0.94258	0.94638	0.93536
18	0.97977	0.97453	0.99634	0.98998	0.85226	0.63850
19	0.96628	0.96097	0.99002	0.99272	0.91926	0.86964
20	0.98631	0.98169	0.97971	0.95914	0.94242	0.91234
21	0.98708	0.97760	0.98939	0.97928	0.95504	0.94054
22	0.99145	0.98747	0.99111	0.96448	0.90074	0.85908
23	0.99391	0.98614	0.98325	0.98912	0.94938	0.91076
24	0.98912	0.98902	0.99239	0.96422	0.96178	0.92958
25	0.90524	0.89650	0.96463	0.91256	0.92642	0.88396
26	0.86154	0.85887	0.87943	0.84346	0.76234	0.73782
27	0.91178	0.93407	0.90287	0.79592	0.83082	0.84098
28	0.88261	0.87920	0.89901	0.93832	0.81372	0.71792
29	0.98055	0.98507	0.92300	0.89790	0.80930	0.76686
30	0.95856	0.94897	0.93700	0.93106	0.81390	0.80098
31	0.99604	0.99028	0.99899	0.99964	0.98006	0.96314
32	0.98061	0.98122	0.95802	0.94178	0.90500	0.87796
33	0.97399	0.96602	0.97599	0.95766	0.93116	0.90708
34	0.99356	0.98958	0.99164	0.94522	0.92062	0.89656
35	0.90270	0.86283	0.92857	0.87956	0.70992	0.61772
36	0.98176	0.97069	0.98948	0.93214	0.90228	0.87042
37	0.99873	0.99707	0.99911	0.96974	0.98362	0.96498
38	0.99488	0.98635	0.99702	0.99290	0.95578	0.89508
39	0.97015	0.95877	0.95811	0.97590	0.92448	0.89638
40	0.96546	0.96143	0.95883	0.93078	0.89306	0.90874
41	0.89218	0.90912	0.85444	0.80208	0.70720	0.76250
42	0.99264	0.99112	0.99190	0.97140	0.95162	0.93066
43	0.92320	0.91166	0.98007	0.93964	0.55174	0.52548
44	0.98860	0.98114	0.98077	0.95566	0.90678	0.89028
45	0.96613	0.96170	0.97195	0.94778	0.94792	0.95978
46	0.99974	0.99930	0.99831	0.99286	0.99016	0.97922
47	0.99926	0.99910	0.99904	0.99086	0.99130	0.97598
48	0.97999	0.96864	0.97639	0.95596	0.96432	0.96334
49	0.99996	0.99994	0.99900	0.99504	0.99894	0.99572
50	0.99533	0.99344	0.98778	0.94522	0.93800	0.93418
51	0.98853	0.98566	0.98125	0.95238	0.92136	0.93976
Total	49.50412	49.28132	49.43454	48.25168	46.03375	44.84610
Wins	20	8	18	5	0	0

TABLE 3: The result of AUC for six algorithms in Dataset number 2. The highest value of AUC result is in bold.

Subject	Algorithm					
	SADD	Sequence alignment	Median vector proximity	Mahalanobis	Manhattan	Euclidean
1	0.88624	0.90732	0.87691	0.84844	0.82584	0.79356
2	0.95560	0.95577	0.93895	0.88018	0.86560	0.85034
3	0.98590	0.97993	0.96152	0.96926	0.90524	0.85016
4	0.99222	0.99102	0.99439	0.97216	0.96684	0.93098
5	0.94865	0.96462	0.94441	0.96218	0.93528	0.95348
6	0.98901	0.99115	0.98706	0.97972	0.95880	0.93774
7	0.99928	0.99776	0.99636	0.98944	0.99494	0.98914
8	0.99156	0.98971	0.98889	0.96762	0.96574	0.96284
9	0.99926	0.99839	0.99928	0.97772	0.98964	0.95676
10	0.99645	0.99884	0.98651	0.96102	0.96532	0.94166
11	0.98800	0.98809	0.95766	0.93818	0.88172	0.85402
12	0.91010	0.87188	0.88252	0.91010	0.82332	0.81468
13	0.99851	0.99646	0.99836	0.99184	0.98646	0.97188
14	0.98852	0.98281	0.98131	0.92130	0.90544	0.88206
15	0.99350	0.99477	0.98994	0.97628	0.96714	0.93850
16	0.98241	0.97288	0.95340	0.93130	0.85616	0.84848
17	0.95874	0.96397	0.94673	0.94258	0.94676	0.93334
18	0.99080	0.98800	0.99653	0.98998	0.90300	0.68836
19	0.97484	0.97116	0.99244	0.99272	0.94462	0.87930
20	0.98398	0.98143	0.97688	0.95914	0.94462	0.91040
21	0.99048	0.98306	0.98878	0.97928	0.96214	0.94290
22	0.98967	0.98849	0.98830	0.96448	0.92518	0.86614
23	0.99664	0.98886	0.97813	0.98912	0.96066	0.91082
24	0.98587	0.98621	0.98557	0.96422	0.96584	0.93000
25	0.91843	0.90906	0.95942	0.91256	0.92510	0.86990
26	0.84544	0.85266	0.85673	0.84346	0.76024	0.73304
27	0.87100	0.93043	0.82663	0.79592	0.81164	0.83336
28	0.93937	0.93577	0.93986	0.93832	0.82262	0.6955
29	0.97335	0.98495	0.91598	0.89790	0.82654	0.77378
30	0.94988	0.94329	0.89873	0.93106	0.81544	0.78808
31	0.99886	0.99779	0.99926	0.99964	0.98600	0.96808
32	0.97698	0.98168	0.94639	0.94178	0.92584	0.88910
33	0.98258	0.97630	0.97367	0.95766	0.93922	0.90810
34	0.99577	0.99215	0.99290	0.94522	0.93410	0.90136
35	0.95729	0.92842	0.96261	0.87956	0.75252	0.61538
36	0.99026	0.98381	0.99333	0.93214	0.92598	0.87978
37	0.99893	0.99773	0.99888	0.96974	0.98948	0.96940
38	0.99652	0.99269	0.99730	0.99290	0.96702	0.90490
39	0.97834	0.97180	0.95770	0.97590	0.94078	0.90424
40	0.97330	0.96871	0.94501	0.93078	0.91412	0.91472
41	0.83326	0.88258	0.78742	0.80208	0.68962	0.74806
42	0.99272	0.99254	0.99048	0.97140	0.95918	0.93112
43	0.95977	0.94853	0.98591	0.93964	0.60102	0.52690
44	0.98470	0.97880	0.97269	0.95566	0.92728	0.89918
45	0.94854	0.95723	0.94442	0.94778	0.94986	0.95918
46	0.99973	0.99959	0.99874	0.99286	0.99284	0.98012
47	0.99938	0.99902	0.99711	0.99086	0.99500	0.97770
48	0.96373	0.96001	0.94336	0.95596	0.96554	0.96292
49	0.99758	0.99742	0.99736	0.99504	0.99928	0.99722
50	0.99167	0.98757	0.97627	0.94522	0.94132	0.93674
51	0.98904	0.98974	0.97148	0.95238	0.92646	0.93746
Total	49.48265	49.47285	49.02047	48.25168	46.53034	44.94286
Wins	23	14	10	2	1	1

TABLE 4: The result of AUC for six algorithms in Dataset number 3. The highest value of AUC result is in bold.

Subject	Algorithm					
	SADD	Sequence alignment	Median vector proximity	Mahalanobis	Manhattan	Euclidean
1	0.91878	0.91561	0.92803	0.84844	0.82134	0.79844
2	0.97017	0.96301	0.97068	0.88018	0.88380	0.85948
3	0.98415	0.97933	0.95722	0.96926	0.90408	0.85802
4	0.99602	0.99316	0.99697	0.97216	0.96790	0.94638
5	0.98845	0.98843	0.99180	0.96218	0.95524	0.96498
6	0.99111	0.98988	0.99043	0.97972	0.92956	0.92516
7	0.99928	0.99884	0.99874	0.98944	0.99544	0.99210
8	0.99434	0.99422	0.99510	0.96762	0.95000	0.94710
9	0.99877	0.99693	0.99721	0.97772	0.95844	0.92222
10	0.99970	0.99974	0.99689	0.96102	0.96640	0.94618
11	0.98862	0.98864	0.96965	0.93818	0.87660	0.86392
12	0.89050	0.86445	0.84490	0.91010	0.82128	0.83304
13	0.99877	0.99702	0.99727	0.99184	0.97992	0.96996
14	0.98520	0.98127	0.97807	0.92130	0.88548	0.87208
15	0.99006	0.99071	0.99299	0.97628	0.94972	0.93500
16	0.95655	0.94369	0.94444	0.93130	0.80648	0.83034
17	0.96680	0.96742	0.97450	0.94258	0.95020	0.93990
18	0.96225	0.95484	0.98715	0.98998	0.80096	0.60398
19	0.95516	0.95233	0.98263	0.99272	0.90200	0.86606
20	0.98398	0.98131	0.97650	0.95914	0.94202	0.91498
21	0.97955	0.97219	0.98599	0.97928	0.95442	0.94144
22	0.99033	0.98696	0.99062	0.96448	0.89646	0.86752
23	0.99109	0.98497	0.98092	0.98912	0.94574	0.91436
24	0.99158	0.99109	0.99410	0.96422	0.95720	0.92892
25	0.87958	0.87253	0.96272	0.91256	0.93030	0.89172
26	0.87085	0.87509	0.88064	0.84346	0.77044	0.74542
27	0.92324	0.93092	0.92630	0.79592	0.84516	0.84584
28	0.86118	0.87980	0.88506	0.93832	0.85035	0.77736
29	0.97665	0.98056	0.92037	0.89790	0.80344	0.76698
30	0.95948	0.95189	0.95176	0.93106	0.82536	0.81706
31	0.98530	0.97686	0.99673	0.99964	0.97476	0.95788
32	0.97732	0.97706	0.95275	0.94178	0.88974	0.87134
33	0.97796	0.97639	0.97776	0.95766	0.93554	0.91100
34	0.99097	0.98795	0.98872	0.94522	0.91766	0.90142
35	0.86485	0.84880	0.88478	0.87956	0.73278	0.64826
36	0.9760	0.96774	0.97859	0.93214	0.90458	0.88226
37	0.99892	0.99692	0.99944	0.96974	0.98150	0.96516
38	0.98589	0.97523	0.99420	0.99290	0.94534	0.88950
39	0.97196	0.96555	0.94897	0.97590	0.91350	0.89376
40	0.96815	0.96767	0.96202	0.93078	0.88326	0.90798
41	0.91268	0.92235	0.88391	0.80208	0.72702	0.77560
42	0.99197	0.99096	0.99086	0.97140	0.95082	0.93314
43	0.89182	0.89006	0.96234	0.93964	0.54252	0.54492
44	0.98709	0.98154	0.98108	0.95566	0.89562	0.88752
45	0.97411	0.97339	0.98341	0.94778	0.94900	0.96054
46	0.99971	0.99942	0.99838	0.99286	0.98820	0.97910
47	0.99952	0.99957	0.99873	0.99086	0.98826	0.97440
48	0.98405	0.97524	0.98639	0.95596	0.96146	0.96468
49	0.99996	0.99994	0.99991	0.99504	0.99734	0.99464
50	0.99212	0.99190	0.98780	0.94522	0.93488	0.92860
51	0.98937	0.98596	0.98243	0.95238	0.92398	0.94170
Total	49.36191	49.21733	49.38885	48.25168	45.96349	44.99934
Wins	20	6	19	6	0	0

TABLE 5: The average of equal error rates with the standard deviation of equal error rate (after plus-minus sign) by using different statistical metrics in both sequence alignment algorithm and SADD. The highest performance of the algorithm and statistical metric is in bold.

Statistical metric	SADD			Sequence alignment		
	Dataset number 1	Dataset number 2	Dataset number 3	Dataset number 1	Dataset number 2	Dataset number 3
Minimum	0.273 ± 0.092	0.335 ± 0.079	0.264 ± 0.096	0.400 ± 0.073	0.472 ± 0.037	0.325 ± 0.106
Maximum	0.083 ± 0.061	0.092 ± 0.059	0.084 ± 0.067	0.088 ± 0.061	0.098 ± 0.056	0.086 ± 0.065
Mean	0.076 ± 0.060	0.078 ± 0.060	0.082 ± 0.062	0.084 ± 0.061	0.079 ± 0.053	0.087 ± 0.061
Median	0.087 ± 0.061	0.095 ± 0.064	0.091 ± 0.063	0.102 ± 0.065	0.115 ± 0.062	0.099 ± 0.064
Mode	0.134 ± 0.076	0.135 ± 0.070	0.146 ± 0.073	0.157 ± 0.078	0.153 ± 0.070	0.160 ± 0.078

TABLE 6: The average of equal error rates with the standard deviation of equal error rate (after plus-minus sign) by using different number of bins in both sequence alignment algorithm and SADD. The highest performance of the algorithms in appropriate number of the bins is in bold.

Bins	Dataset number 1		Dataset number 2		Dataset number 3	
	SADD	SA	SADD	SA	SADD	SA
10	0.083 ± 0.070	0.089 ± 0.073	0.079 ± 0.066	0.077 ± 0.064	0.095 ± 0.079	0.101 ± 0.082
20	0.076 ± 0.060	0.084 ± 0.061	0.078 ± 0.060	0.079 ± 0.053	0.082 ± 0.062	0.087 ± 0.061
30	0.077 ± 0.059	0.084 ± 0.059	0.082 ± 0.068	0.083 ± 0.061	0.076 ± 0.057	0.084 ± 0.060
40	0.077 ± 0.060	0.086 ± 0.061	0.082 ± 0.067	0.087 ± 0.062	0.079 ± 0.059	0.084 ± 0.061
50	0.079 ± 0.059	0.086 ± 0.062	0.084 ± 0.068	0.087 ± 0.062	0.079 ± 0.060	0.085 ± 0.063

age, and the dominant hand (i.e., left or right). This extra information can be used as reference in order to help to improve the performance of algorithms. Although we do not include this extra information into our experiment due to the limitation information that we can obtain from the CMU benchmark dataset, we believe that this extra information will be beneficial in the future work.

Syed et al. [8] show the concept of event sequences used in the keystroke dynamics. These event sequences help in distinguishing the typing behavior of a user. Most keystroke dynamics use key-down and key-up without actual key values. However, introducing too many dimensions can cause the curse of dimensionality. Therefore, it is interesting to extend our algorithm to accommodate these event sequences in the future work.

7. Conclusion

In this paper, we have proposed sequence alignment with dynamic divisor generation (SADD) for user authentication by using the keystroke dynamics. Based on the experiments we have conducted, our algorithm produces promising results and also mostly outperforms other previous work. We also empirically show that the dynamic divisor generally outperforms static divisor. We believe that the dynamic divisor takes an important role in sequence alignment algorithm because it calculates the degree of sufficiency of the dataset (by using mean of Horner's rule) and then it provides faultless calculation for an appropriate divisor to be used in each attribute. These dynamic divisors help to prevent the genuine user's data digressed from the legal categories.

Based on Giot and Rosenberger's [3] research, they introduce a new soft biometric for keystroke dynamics based on gender recognition. They have done interesting work by

introducing this new information in keystroke dynamics. Furthermore, Idrus et al. [4] also introduce more valuable information such as the type of hands used (i.e., one hand or both hands), age, and the dominant hand (i.e., left or right). However, they just study the effectiveness of using the information. Therefore, it will be interesting to apply this extra information to our future study. We also like to discover more valuable information besides the information we have discussed above.

Conflict of Interests

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

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (no. NRF-2013R1A1A2013401).

References

- [1] J. C. Poss, D. Boye, and M. W. Mobley, "Biometric voice authentication," Patent and Trademark Office, Washington, DC, USA, U.S. Patent no. 7,386,448, 2008.
- [2] S. Cho, C. Han, D. H. Han, and H.-I. Kim, "Web-based keystroke dynamics identity verification using neural network," *Journal of Organizational Computing and Electronic Commerce*, vol. 10, no. 4, pp. 295–307, 2000.
- [3] R. Giot and C. Rosenberger, "A new soft biometric approach for keystroke dynamics based on gender recognition," *International Journal of Information Technology and Management*, vol. 11, no. 1-2, pp. 35–49, 2012.

- [4] S. Z. S. Idrus, E. Cherrier, C. Rosenberger, and P. Bours, "Soft biometrics for keystroke dynamics," in *Image Analysis and Recognition*, vol. 7950 of *Lecture Notes in Computer Science*, pp. 11–18, Springer, Berlin, Germany, 2013.
- [5] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *Proceedings of the IEEE/IFIP International Conference on Dependable Systems & Networks (DSN '09)*, pp. 125–134, IEEE, July 2009.
- [6] M. M. Al-Jarrah, "An anomaly detector for keystroke dynamics based on medians vector proximity," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 3, no. 6, pp. 988–993, 2012.
- [7] K. Revett, "A bioinformatics based approach to user authentication via keystroke dynamics," *International Journal of Control, Automation and Systems*, vol. 7, no. 1, pp. 7–15, 2009.
- [8] Z. Syed, S. Banerjee, and B. Cukic, "Leveraging variations in event sequences in keystroke-dynamics authentication systems," in *Proceedings of the IEEE 15th International Symposium on High-Assurance Systems Engineering (HASE '14)*, pp. 9–16, IEEE, January 2014.
- [9] X. Wang, F. Guo, and J.-F. Ma, "User authentication via keystroke dynamics based on difference subspace and slope correlation degree," *Digital Signal Processing: A Review Journal*, vol. 22, no. 5, pp. 707–712, 2012.
- [10] E. Yu and S. Cho, "Keystroke dynamics identity verification—its problems and practical solutions," *Computers and Security*, vol. 23, no. 5, pp. 428–440, 2004.
- [11] R. Moskovitch, C. Feher, A. Messerman et al., "Identity theft, computers and behavioral biometrics," in *Proceedings of the IEEE International Conference on Intelligence and Security Informatics (ISI '09)*, pp. 155–160, June 2009.
- [12] A. F. M. N. H. Nahin, J. M. Alam, H. Mahmud, and K. Hasan, "Identifying emotion by keystroke dynamics and text pattern analysis," *Behaviour & Information Technology*, vol. 33, no. 9, pp. 987–996, 2014.
- [13] N. B. Amor, S. Benferhat, and Z. Elouedi, "Naive Bayes vs decision trees in intrusion detection systems," in *Proceedings of the ACM Symposium on Applied Computing*, pp. 420–424, ACM, March 2004.
- [14] R. Fabbri, L. D. F. Costa, J. C. Torelli, and O. M. Bruno, "2D Euclidean distance transform algorithms: a comparative survey," *ACM Computing Surveys*, vol. 40, no. 1, article 2, 2008.
- [15] R. Koch, M. Golling, and G. D. Rodosek, "Behavior-based intrusion detection in encrypted environments," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 124–131, 2014.
- [16] P. O. Asagba and E. O. Nwachukwu, "RSA asymmetric cryptosystem beyond homogeneous transformation," *West African Journal of Industrial and Academic Research*, vol. 9, no. 1, pp. 3–12, 2014.
- [17] D. W. Mount, *Bioinformatics: Sequence and Genome Analysis*, Cold Spring Harbour, Cold Spring Harbour Laboratory Press, 2nd edition, 2004.
- [18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, 3rd edition, 2009.

Research Article

Distributed Software-Attestation Defense against Sensor Worm Propagation

Jun-Won Ho

Department of Information Security, Seoul Women's University, Seoul 139-774, Republic of Korea

Correspondence should be addressed to Jun-Won Ho; jwho@swu.ac.kr

Received 15 November 2014; Accepted 6 March 2015

Academic Editor: Fanli Meng

Copyright © 2015 Jun-Won Ho. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks are vulnerable to sensor worm attacks in which the attacker compromises a few nodes and makes these compromised nodes initiate worm spread over the network, targeting the worm infection of the whole nodes in the network. Several defense mechanisms have been proposed to prevent worm propagation in wireless sensor networks. Although these proposed schemes use software diversity technique for worm propagation prevention under the belief that different software versions do not have common vulnerability, they have fundamental drawback in which it is difficult to realize the aforementioned belief in sensor nodes. To resolve this problem, we propose on-demand software-attestation based scheme to defend against worm propagation in sensor network. The main idea of our proposed scheme is to perform software attestations against sensor nodes in on-demand manner and detect the infected nodes by worm, resulting in worm propagation block in the network. Through analysis, we show that our proposed scheme defends against worm propagation in efficient and robust manner. Through simulation, we demonstrate that our proposed scheme stops worm propagation at the reasonable overhead while preventing a majority of sensor nodes from being infected by worm.

1. Introduction

Since sensor nodes are usually deployed in unattended manner, they could be physically captured by an attacker. By compromising the captured nodes and launching various attacks with these compromised nodes, an attacker can have a malicious impact on the sensor networks. However, it takes substantial amount of time to physically capture and compromise a large number of nodes and thus it is not suitable for fast-spread node compromise in sensor networks. To mitigate this limitation, an attacker could launch worm attacks in which self-propagating worm originates from a few number of compromised nodes and is widely spread over the network in order to convert the benign nodes to the malicious nodes. Because worm could be very quickly propagated over the network, it is very important to detect and stop worm propagation as soon as possible in order to minimize the number of nodes infected by worm.

To meet this goal, several schemes have been developed for worm defense in the sensor network [1–3]. The key idea of these schemes prevents worm propagation by enforcing

two adjacent nodes to have distinct software versions, leading to worm dissemination failure between two adjacent nodes. However, if different software versions can have common vulnerability, these schemes will not work because the shared vulnerability can be exploited for worm to spread between two adjacent nodes. Therefore, more active defense mechanisms without this drawback should be required rather than passive techniques such as software diversity.

To achieve this requirement, we propose on-demand software-attestation based scheme to actively defend worm propagation in sensor network. Software attestation is a malicious node detection technique in which the flash image of sensor node is checked with the normal image and the subverted flash image is detected [4–8]. We adapt software attestation to actively defend the network against worm attacks. More specifically, every node randomly chooses a list of attestees against which it is in charge of attesting. Whenever it receives packets from the neighboring nodes that belong to the attestee list, it performs software attestations against these neighbors and detects the nodes infected by worm, leading to hindrance of worm spread over the network.

We analytically show that our proposed scheme hampers sensor worm propagation robustly and efficiently. In addition, we evaluate the performance of our proposed scheme through simulation. The simulation results illustrate that at most 3655.7 attestations per time slot are required for worm propagation barrier on average and at most 54.5 attestations per time slot are required for benign traffic on average, where the network consists of 1000 nodes. Furthermore, the fraction of infected nodes does not exceed 7.86% and 23.26% when the size of the attestee list stored in a node is 200 and 100, respectively. These results indicate that our proposed scheme restrains a majority of sensor nodes from being infected at the reasonable attestation overhead.

The rest of the paper is organized as follows. In Section 2, we present the related work. In Section 3, we describe the network assumptions and adversary models for our scheme. In Section 4, we propose a worm propagation defense scheme based on software attestation in sensor networks and then present the security and performance analyses of our proposed scheme. In Section 5, we present the simulation results of our proposed scheme. In Section 6, we conclude the paper.

2. Related Work

Recent research work demonstrates that malicious code injection and worm propagation attacks can be launched against sensor devices [9–11]. Goodspeed [10] leveraged format string and buffer overflow vulnerabilities to execute malicious codes on the MSP430-based TelosB motes. Gu and Noorani [11] demonstrated that malicious codes can be temporarily executed on Mica2 motes and spread to neighboring motes. Francillon and Castelluccia [9] showed that malicious codes can be permanently and remotely injected into the program memory of Atmel AVR-based sensor motes such as the MicaZ mote. To the best of our knowledge, this work is the first to shatter a common opinion that permanent malicious code injection is impossible on sensor devices designed in the Harvard architecture.

To defend against the sensor worm attacks, several schemes are proposed to prevent worm propagation in sensor networks [1–3, 12]. Yang et al. [3] first applied a software diversity technique to prevent worm propagation in sensor networks. The key idea of this work is that the flash image of the nodes in a cell A is different to the one of the nodes in the A 's adjacent cells, where the network is broken into a set of grid cells. The proposed scheme will prevent a worm from propagating through two adjacent cells as long as different versions of flash image do not have any common vulnerabilities. However, it is hard to automatically guarantee that different versions of flash image have distinct vulnerabilities. If we can find them out, we can fix them and thus we do not need this scheme. On the other hand, when all versions of flash image have the common vulnerability that has not yet been detected, worm can propagate through two adjacent cells by exploiting this vulnerability, leading to the entire network infection. Gui et al. [1] explored how the software diversity technique defers worm infection through active sensor nodes when they are selected by random node scheduling. Liu et al. [2] proposed a role-based graph coloring scheme for software diversity

in sensor networks. Since both [1, 2] leverage the software diversity technique for worm propagation prevention, they have the same limitations as in [3]. Sun et al. [12] proposed an immunization-based worm defense in which the network operator employs a set of immune nodes to prevent or slow down worm propagation. However, [12] mainly focuses on how to select the immune nodes while not presenting any details of how immune nodes can stop worm propagation.

Software attestation is a useful technique to discern the subverted flash image of malicious node with virtually zero false positives. Several researchers applied software attestation for malicious node detection in sensor networks [4–8]. More specifically, the base station determines whether the flash image has been maliciously altered by attesting randomly chosen portions of flash image or the entire one [5, 6]. In both [4, 7], the attestation process is performed in localized and distributed manner. In [8], the dynamic attestation process is performed against the executing flash image.

However, these schemes do not specify how often the attesters need to perform the attestation against the attestees. Our proposed scheme performs software attestation in on-demand manner in order to block sensor worm propagation in the network.

3. Preliminaries

In this section, we first describe the network assumptions and then the adversary model for our proposed scheme.

3.1. Network Assumptions. We assume a *static* sensor network that is widely adopted for sensor node deployment. In static sensor network, each sensor node never alters its position after being deployed in the network. Furthermore, we assume that a node can directly communicate with one-hop neighboring nodes within its communication vicinity. Through multihop communication, a node is thus able to communicate with nodes out of one-hop communication range.

3.2. Adversary Models. We assume that the attacker can compromise a set of nodes, inject worm codes into these compromised nodes, and make them worm originators, leading to widespread worm propagation. Moreover, we assume that the flash images of the infected nodes are subverted such that these infected nodes perform various malicious activities.

In sensor networks, the normal network operations such as data aggregation, clustering, and time synchronization are generally performed through the local communications rather than arbitrary peer-to-peer ones [13], and accordingly the normal network traffic is usually propagated in a neighbor-to-neighbor manner. In order to minimize the chance of being detected, the attacker would like to have worm propagation pattern be similar to the ordinary pattern of the normal network traffic in sensor networks. From this perspective, it is reasonable to assume that the attacker adopts a hop-by-hop worm propagation strategy in which worm is propagated in a neighbor-to-neighbor manner.

Although hop-by-hop propagation strategy indicates how worm is spread over the sensor network, it does not specify how many nodes can be infected by worm within a certain

amount of time. To quantify the worm infection quota, we use the discrete time version of simple epidemic model [14], which is widely employed to model worm propagation in wired and wireless networks. In simple epidemic model, a sensor node is assumed to have two states: *susceptible* and *infectious*. All sensor nodes besides worm originators, which are in infectious state, are initialized to the susceptible state. If the susceptible nodes are infected by worm, their state will be switched to infectious. Furthermore, the infection quota in units of time slots is given by

$$I_t = (1 + \epsilon n) I_{t-1} - \epsilon I_{t-1}^2, \quad (1)$$

where n is the total number of sensor nodes in the network and ϵ is the pairwise infection rate [14]. I_0 represents the number of worm originators. I_t is the cumulative infection quota from the 0th time slot to the t th time slot. Therefore, the infection quota in the t th time slot is computed as $I_t - I_{t-1}$.

In the simple epidemic model together with hop-by-hop propagation strategy, the network topology could make the infectious nodes have fewer susceptible neighbors than the infection quota. As a result, the worm infection quota might not be fulfilled in a time slot. To soothe this problem, we slightly modify the discrete time version of simple epidemic model such that the infection quota deficit in the current time slot is carried over to the next time slot, meeting the total infection quota over the entire time period. In the modified version of simple epidemic model, the infection quota in the $t + 1$ st time slot is calculated as $I_{t+1} - I_t + b_t$ ($t \geq 1$), where b_t is the infection quota deficit in the t th time slot and the infection quota in the first time slot is $I_1 - I_0$.

4. Sensor Worm Propagation Defense Using On-Demand Software Attestation

In this section, we first describe the details of our sensor worm propagation defense scheme and then analyze the security and performance of it.

4.1. Protocol Description. Let us denote n as the total number of nodes in the network. Each node has distinct ID and accordingly there are n distinct IDs in the network. Moreover, we assume that the entire time domain is divided into a series of time slots. After being deployed, every node acts as *attester* during its lifetime. As attester, each node selects the list of *attestees* against which it performs attestations. More specifically, each attester generates the attestee list by selecting m distinct IDs uniformly at random from the entire ID space (i.e., n distinct IDs). Note that the attestee selection process is repeated every time slot and thus a node will highly likely have distinct attestee list for each time slot.

Suppose that a node u sends a packet to a node v where u and v are in the vicinity of each other. Upon receiving the packet sent by u , v first checks whether u belongs to the attestee list. If so, it works as primary-attester and performs software attestations against u . If the attester v checks whether the flash image of u has been subverted and decides the attestee u as infectious node, it sends *Attester Role Assignment* message to all neighboring nodes. Upon receiving this message, each neighbor w of v works as secondary-attester such

that it performs the attestations against all neighboring nodes each time it receives packets from them. Note that the attester v will work as secondary-attester as well as primary-attester in order to expedite the infectious node detection.

In the sense that the infectious nodes are highly likely close to each other due to hop-by-hop worm propagation strategy, the attester role assignment contributes to fast detection of infectious nodes.

After detecting infectious neighbor nodes, every node stops communicating with them and thus the infectious nodes are isolated from the network.

4.2. Security Analysis. In this section, we derive the probability that an infectious node is detected by primary-attesters.

Recall that there are n distinct IDs in the network and a node maintains the list of m attestees against which it performs attestations. When a primary-attester receives a worm packet from an infectious node, the probability that it detects an infectious node is calculated as m/n . Therefore, when s primary-attesters receive a worm packet from an infectious node, the probability that they detect an infectious node is given by

$$\begin{aligned} P_s &= 1 - \prod_{i=1}^s \left(1 - \frac{m}{n}\right) \\ &= 1 - \left(1 - \frac{m}{n}\right)^s. \end{aligned} \quad (2)$$

By applying the fact that $(1 + x) \leq e^x$ to (2), we have

$$\begin{aligned} P_s &= 1 - \left(1 - \frac{m}{n}\right)^s \\ &\geq 1 - e^{-ms/n}. \end{aligned} \quad (3)$$

From (3), the lower bound on P_s is calculated as $1 - e^{-ms/n}$.

Figure 1 shows how the lower bound of infectious node detection probability is affected by the number of primary-attesters and the ratio of attestee list size to total number of nodes. When $m/n \geq 0.25$, it is guaranteed that an infectious node is detected with probability of more than 0.95. Furthermore, when the number of primary-attesters is at least 30, infectious detection probability is at least 0.95 even in case of $m/n = 0.1$. This demonstrates that our proposed scheme achieves high infectious node detection capability. As s increases, the lower bound on P_s tends to rise. This indicates that a growth in the number of primary-attesters contributes to an increase in the likelihood that infectious node is detected. Moreover, as s rises, we observe the narrower gap among the lower bounds on P_s in three cases of m/n . This implies that the growth rate in lower bound on P_s rises as m/n falls off.

4.3. Performance Analysis. In this section, we compute the attestation, communication, and storage overhead of our proposed scheme. For this purpose, we take account of *benign* and *worm* case. In benign case, we consider $R \geq 1$ distinct sender-receiver pairs such that each sender sends $\lambda \geq 1$ packets to each receiver. In worm case, we consider

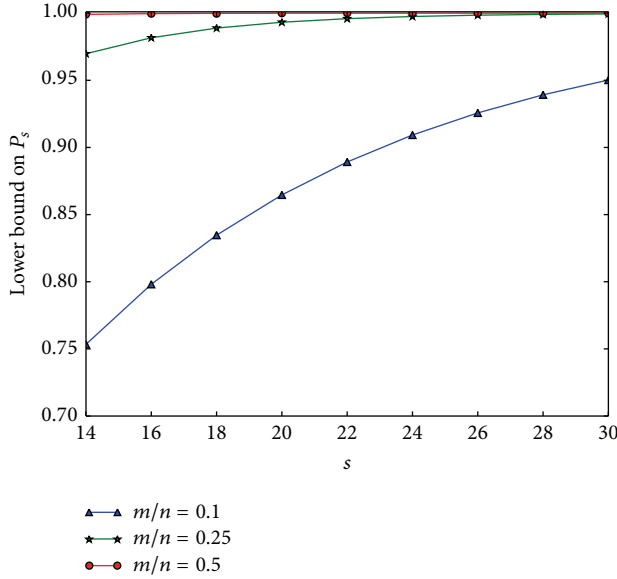


FIGURE 1: The effects of s and m/n on the lower bound of infectious node detection probability.

R distinct infectious-susceptible pairs such that an infectious node propagates λ worm packets to a susceptible node. We also assume that each node has d neighbors on average.

We define the attestation overhead as an average number of software attestations that are performed in the network. When there is no traffic in the network, the attestation overhead will be zero because each node initiates attestation process against a neighbor node only if it receives packets from that neighbor. In benign case, the attestation overhead is calculated as $R \times \lambda \times (m/n)$ since a receiver is a primary-attester and it attests against a sender with probability m/n whenever it receives a benign packet. In worm case, the attestation overhead incurred by primary-attesters is $R \times \lambda \times (m/n)$ because R susceptible nodes are primary-attesters. Since all neighbors of primary-attesters act as secondary-attester and primary-attesters also take secondary-attester role, there are at most $R \times (d + 1)$ secondary-attesters in the network. Let us consider a worst-case scenario in which all secondary-attesters are susceptible and each secondary-attester receives $\lambda' \geq 1$ worm packets. The attestation overhead incurred by secondary-attesters is at most $R \times (d + 1) \times \lambda' \times (m/n)$ in the worse case. As a consequence, the total attestation overhead in the worst case is calculated as $R \times (m/n) \times ((d + 1)\lambda' + \lambda)$.

We define the communication overhead as an average number of *Attester Role Assignment* messages sent in the network. In benign case, communication overhead is zero because there are no infectious nodes and thus *Attester Role Assignment* messages are never sent by attesters. In worm case, after primary-attesters detect infectious nodes by attesting against them, they send *Attester Role Assignment* messages to their neighbors. Thus, the communication overhead is computed as $R \times (m/n) \times d$.

Finally, we define storage overhead as the number of IDs that need to be stored per node. Since each node stores m IDs that are randomly selected, this overhead is calculated as m in both benign and worm cases.

5. Simulation Study

In this section, we first explain the simulation environments and then describe the simulation results.

5.1. Simulation Environment. We developed a simple simulation program to evaluate our proposed scheme. In our simulation, we place 1000 sensor nodes in a square area field of $1000 \text{ m} \times 1000 \text{ m}$ and configure the communication radius of a sensor node to 50 m. Under this network setting, we are able to evaluate how our proposed scheme detects worm propagation in large-scale network. Moreover, we employ a group deployment strategy in which a group of sensor nodes is placed toward the group deployment point and the actual placement follows the two-dimensional Gaussian distribution:

$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-((x-x_g)^2 + (y-y_g)^2)/2\sigma^2}, \quad (4)$$

where (x_g, y_g) is the group deployment point and σ is the standard deviation. In our group deployment, the number of groups is set to 20, the number of nodes in a group is configured to 50, and σ is set to 50. We set $m = 100, 200$, where m is the size of the attestee list that a node needs to store.

We evaluate our scheme in two cases: *benign* and *worm*. In the benign case, we consider only benign traffic in the network. In the worm case, we consider only worm traffic in the network. The main rationale behind this case separation is to explore the worm spread defense capability and the attestation overhead without being affected by the benign packets and the worm packets, respectively.

In the benign case, we model the number of benign packets to be generated as a homogeneous Poisson process with rate parameter γ . As a consequence, the interarrival times of benign packets follow the exponential distribution. More specifically, the interarrival time between two consecutive benign packets is calculated as $-(\ln(U))/\gamma$, where U is uniform random variate such that $0 \leq U < 1$. We configure γ from 1.0 to 10.0 via increments of 1.0. Under these settings, the number of benign packets to be generated is from 3000 to 30000 via increase of 3000 on average. For each time slot, we randomly select as many pairs of source and destination nodes as the number of benign packets to be generated by Poisson process. Additionally, destination node is randomly chosen from the neighbors of source node. This is reasonable because local communication patterns are more prevalent than arbitrary peer-to-peer ones in sensor networks [13].

In the worm case, as described in Section 3, we adopt a hop-by-hop worm propagation strategy together with the discrete-time version of the simple epidemic model [14], in which the infection quota deficit in the current time slot is carried over to the next time slot, meeting the total infection quota over the entire time period. Recall that ϵ is the pairwise infection rate; I_0 indicates the number of worm originators. We consider a single worm originator and accordingly I_0 is set to one. We set ϵ from 0.0001 to 0.001 via increments of 0.0001. Furthermore, we assume that a single worm packet is used to infect a susceptible node. This assumption is regarded as the best-case scenario for the attacker because the fast

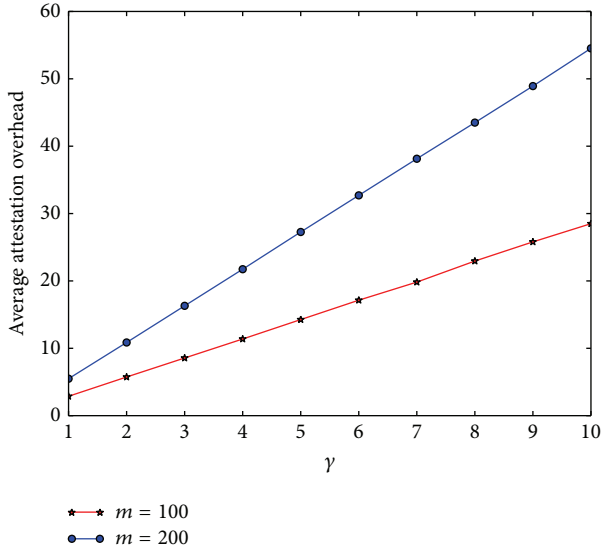


FIGURE 2: The effect of Poisson rate parameter γ on the attestation overhead in the benign case.

infection is beneficial for the attacker. Hence, we evaluate our scheme under the worst-case scenario in terms of the number of worm packets.

5.2. *Simulation Results.* We use the following metrics to evaluate the performance of our proposed scheme.

- (i) *Fraction of Infectious Nodes* is the fraction of infectious nodes when all worm propagations are blocked in the network.
- (ii) *Attestation Overhead* is the number of attestations that are performed per time slot in the network.
- (iii) *Communication Overhead* is the number of *Attester Role Assignment* messages that are sent per time slot in the network.

For each metric, we present the average results for 100 runs of the simulation in each configuration such that each run is executed for 100 time slots, where a time slot duration is 30 simulation seconds. Our main findings are as follows.

In the benign case, as shown in Figure 2, all nodes perform at most 54.5 attestations per time slot on average. This means that a small number of attestations are performed in the network. Furthermore, attestation overhead rises as γ increases in the benign case. This is because the more benign traffic incurs the higher attestation overhead. Given a value of γ , a rise in m contributes to an increase in attestation overhead. This is because more attestations are performed as the size of attestee list grows. In particular, as γ rises, we observe the wider gap between the attestation overhead in case of $m = 100$ and the one in case of $m = 200$. This implies that the attestee list needs to be kept in small size in order to reduce the attestation overhead under the high benign traffic.

Figures 3 and 4 show how attestation and communication overhead are affected by ϵ and m in the worm case. In terms of attestation overhead, all nodes perform at most 3655.7 and 1420.4 attestations for each time slot on average when

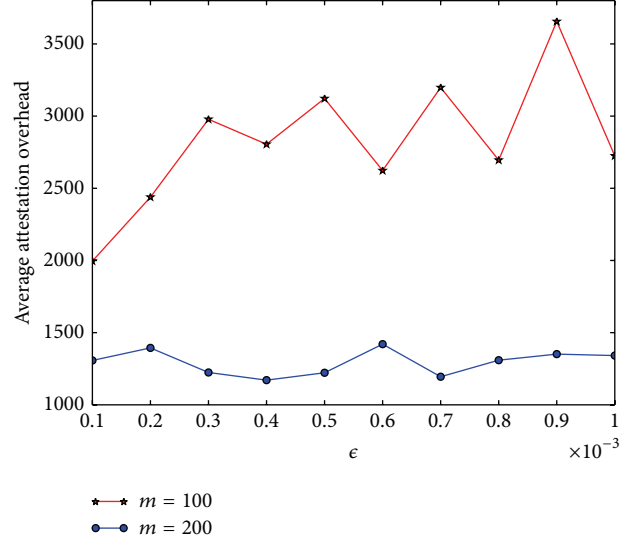


FIGURE 3: The effect of worm infection rate ϵ on the attestation overhead in the worm case.

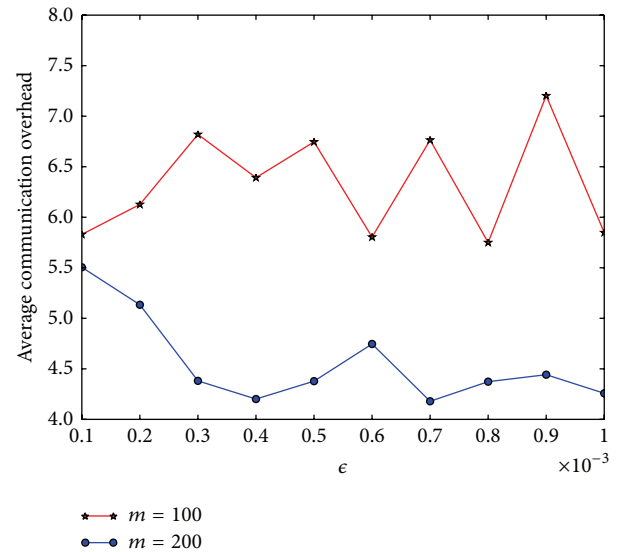


FIGURE 4: The effect of worm infection rate ϵ on the communication overhead in the worm case.

$m = 100$ and $m = 200$, respectively. This represents the fact that a node performs below four attestations per time slot on average, burdening insignificant overhead on each node. In terms of communication overhead, the number of *Attester Role Assignment* messages sent per time slot is at most 7.2 on average, incurring slight overhead in the network. Both attestation and communication overhead in case of $m = 200$ are less than the ones in case of $m = 100$. This indicates that the large size of attestee list contributes to diminishing the attestation and communication overhead in all cases of infection rates. Putting it in a different way, when more nodes are attested, worm propagation is blocked at the earlier time, leading to decay in the attestation and communication overhead. We also observe that the attestation overhead more severely fluctuates over ϵ in case of $m = 100$ than $m = 200$. On

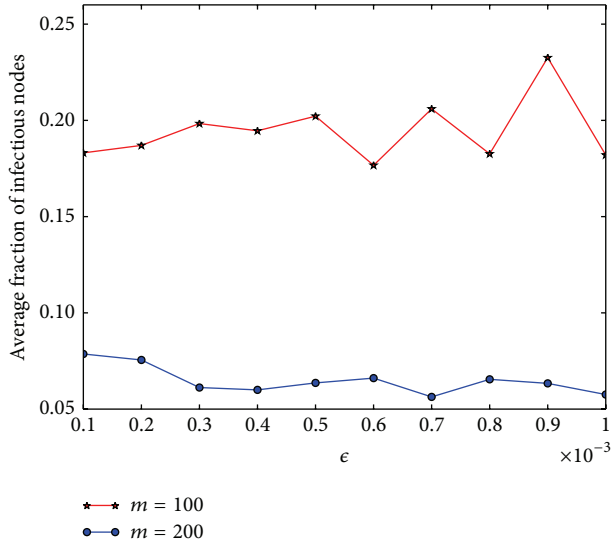


FIGURE 5: The fraction of infectious nodes versus worm infection rate ϵ in the worm case.

the other hand, in both cases, the communication overhead exhibits relatively small fluctuation over ϵ .

As shown in Figure 5, the fraction of infectious nodes is limited to at most 7.86% and 23.26% in case of $m = 200$ and $m = 100$, respectively. This means that our proposed scheme stops worm propagation while minimizing the number of infectious nodes. Moreover, the fraction of infectious nodes in case of $m = 200$ is substantially lower than the one in case of $m = 100$. This indicates that more attestations result in the lower fraction of infectious nodes.

6. Conclusions

In this paper, we proposed on-demand software-attestation based scheme to stop worm propagation in sensor network. We also analytically showed that our proposed scheme blocks sensor worm propagation in efficient and robust manner. Furthermore, we evaluated the proposed scheme through simulation. The simulation results demonstrate that our proposed scheme stops worm propagation with at most 3655.7 attestations per time slot on average while at most 54.5 attestations per time slot are required on average in benign scenario, where 1000 nodes are employed in the network. Moreover, the fraction of infectious nodes is sustained to at most 7.86% and 23.26% when the size of the attessee list that is maintained by a node is 200 and 100, respectively.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by a young researchers grant from Seoul Women's University (2014).

References

- [1] N. Gui, E. Zhai, J. Hu, and Z. Chen, "SWORDS: improving sensor networks immunity under worm attacks," in *Web-Age Information Management: 11th International Conference, WAIM 2010, Jiuzhaigou, China, July 15–17, 2010. Proceedings*, vol. 6184 of *Lecture Notes in Computer Science*, pp. 86–96, Springer, Berlin, Germany, 2010.
- [2] Y. Liu, W. Zhang, S. Bai, and C. Wang, "Defending sensor worm attack using software diversity approach," in *Proceedings of the IEEE International Conference on Communications (ICC '11)*, pp. 1–5, IEEE, June 2011.
- [3] Y. Yang, S. Zhu, and G. Cao, "Improving sensor network immunity under worm attacks: a software diversity approach," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, pp. 149–158, May 2008.
- [4] T. Abuhmed, J. Kang, D. Nyang, and K. H. Lee, "A software-based group attestation for wireless sensor networks," *Ad-Hoc & Sensor Wireless Networks*, vol. 13, no. 1-2, pp. 121–154, 2011.
- [5] T. Park and K. G. Shin, "Soft tamper-proofing via program integrity verification in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 4, no. 3, pp. 297–309, 2005.
- [6] A. Seshadri, A. Perrig, L. van Doorn, and P. Khosla, "SWATT: softWare-based attestation for embedded devices," in *Proceedings of the IEEE Symposium on Security and Privacy (S&P '04)*, pp. 272–282, May 2004.
- [7] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Distributed software-based attestation for node compromise detection in sensor networks," in *Proceedings of the 26th IEEE International Symposium on Reliable Distributed Systems (SRDS '07)*, pp. 219–230, IEEE, Beijing, China, October 2007.
- [8] D. Zhang and D. Liu, "DataGuard: dynamic data attestation in wireless sensor networks," in *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN '10)*, pp. 261–270, July 2010.
- [9] A. Francillon and C. Castelluccia, "Code injection attacks on harvard-architecture devices," in *Proceedings of the 15th ACM conference on Computer and Communications Security (CCS '08)*, pp. 15–25, October 2008.
- [10] T. Goodspeed, "Exploiting wireless sensor networks over 802.15.4," in *Proceedings of the Texas Instruments Developer Conference*, November 2008.
- [11] Q. Gu and R. Noorani, "Towards self-propagate mal-packets in sensor networks," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 172–182, Alexandria, Va, USA, April 2008.
- [12] B. Sun, G. Yan, Y. Xiao, and T. Andrew Yang, "Self-propagating mal-packets in wireless sensor networks: dynamics and defense implications," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1489–1500, 2009.
- [13] 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.
- [14] D. J. Daley and J. Gani, *Epidemic Modeling: An Introduction*, vol. 15 of *Cambridge Studies in Mathematical Biology*, Cambridge University Press, Cambridge, UK, 1999.

Research Article

B-iTRS: A Bio-Inspired Trusted Routing Scheme for Wireless Sensor Networks

Mingchuan Zhang,^{1,2} Ruijuan Zheng,¹ Qingtao Wu,¹ Wangyang Wei,¹
Xiuling Bai,¹ and Haixia Zhao¹

¹Information Engineering College, Henan University of Science and Technology, Luoyang 471023, China

²National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Ruijuan Zheng; rjwo@163.com

Received 24 November 2014; Revised 11 March 2015; Accepted 20 March 2015

Academic Editor: Fei Yu

Copyright © 2015 Mingchuan Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In WSNs, routing algorithms need to handle dynamical changes of network topology, extra overhead, energy saving, and other requirements. Therefore, routing in WSNs is an extremely interesting and challenging issue. In this paper, we present a novel bio-inspired trusted routing scheme (B-iTRS) based on ant colony optimization (ACO) and Physarum autonomic optimization (PAO). For trust assessment, B-iTRS monitors neighbors' behavior in real time, receives feedback from Sink, and then assesses neighbors' trusts based on the acquired information. For routing scheme, each node finds routes to the Sink based on ACO and PAO. In the process of path finding, B-iTRS senses the load and trust value of each node and then calculates the link load and link trust of the found routes to support the route selection. Moreover, B-iTRS also assesses the route based on PAO to maintain the route table. Simulation results show how B-iTRS can achieve the effective performance compared to existing state-of-the-art algorithms.

1. Introduction

Mobile wireless sensor networks (WSNs) are autonomous wireless communication networks and ideal candidates for applications to report detected events of interest, such as military surveillance and forest fire [1, 2]. Because of the sensor nodes' mobility and failures, limited bandwidth, and power energy, routing algorithms need to handle dynamical changes of network topology, extra overhead, energy saving, and other requirements. Therefore, routing in WSNs is an extremely interesting and challenging issue.

Traditional WSNs routing protocols assume that all sensor nodes work in a benevolent manner, which may render the WSNs vulnerable to malicious attacks in case of the presence of selfish and malicious nodes. Routing protocols, data, battery power, and bandwidth are the common targets of these attacks. Scientific researches prove that selfish behavior will seriously affect the network performance [3]. Since the safety of multihop communication depends on the reliability of nodes on the route to destination primarily, it is important

for routing protocols to know the reliability of the nodes forming the route. Moreover, how to guarantee the efficiency of the multihop route is important to prolong lifetime of WSNs. Therefore, security and efficiency are the significant features for routing in WSNs.

Recently, many research results on the trust and efficiency of routing have been proposed. In order to solve the security of routing protocol, some techniques (e.g., trust value, detection, cryptography, and data hiding) are proposed based on different applications [1–5]. Bao et al. [4] propose a highly scalable trust-based geographic routing protocol (TGRP) for WSNs to effectively deal with selfish or malicious nodes. Zhan et al. [1] design and implement TARE, a robust trust-aware routing framework for dynamic WSNs. Some methods (e.g., location-aware method, energy-aware method, energy harvesting method, and their combination) are discussed [6–9]. Yang and Heinzelman [7] propose sleeping multipath routing, which selects the minimum number of disjoint paths to achieve the trade-off of given reliability requirement and energy efficiency. Trajcevski et al. [8] present heuristic

approaches to relieve some of the routing load of the boundary nodes of energy holes in location-aware WSNs. Chen et al. [9] present a method to enhance the efficiency of gathering sensor data based on a quadrotor-based mobile Sink.

Bio-inspired methods [10–15] are advantageous for solving the problems regardless of security and efficiency of routing protocols. Gunes et al. [10] present an on-demand routing algorithm ARA based on ant colony optimization (ACO) and AODV [16]. Di Caro et al. [11] propose the hybrid routing algorithm AntHocNet, where artificial ants reactively set up multiple routes on demand and proactively test existing paths and explore new paths during the course of communication session. Tero et al. [12] propose a mathematical model for the Physarum autonomic optimization (PAO). Li et al. [13] present a routing protocol for wireless sensor networks based on PAO. We study the foraging rule of Physarum and present a Physarum-inspired routing protocol for WSNs [14, 15].

In addition, trust computation or management is important for assessing node's reliability. Ren et al. [17] propose a trust management scheme for UWSNs to provide efficient and robust trust data storage and trust generation, where a geographic hash table is employed. Chen et al. [18] design and validate a dynamic trust management protocol for secure routing optimization in DTN environments in the presence of well-behaved, selfish, and malicious nodes. Priyanka et al. [19] present a malicious node detection scheme for wireless sensor networks, where the malicious nodes are detected by computing the average number of event cycles. Indhu Lekha and Kathirolu [20] propose a vector based trust mechanism which nominates a cluster head based on the higher trust value computation with the earliest bit vectors and enhanced certificate revocation scheme for discarding the authorization of the misbehaving nodes. Wang et al. [21] propose the framework ARTSense to solve the problem of "trust without identity" in mobile sensing, where contextual factors are employed to dynamically affect the trustworthiness of the sensing data as well as the mutual support and conflict among data from different sources.

In this paper, we propose a novel bio-inspired trusted routing scheme (B-iTRS) in comprehensive thought of trust and load. B-iTRF consists of trust mechanism and routing scheme. For trust mechanism, B-iTRS assesses nodes' trust value through monitoring nodes' behavior in real time and receives feedback from Sink. For routing scheme, B-iTRS finds routes to the Sink based on ACO by introducing cross-layer [22] and assesses the discovered routes based on PAO.

The rest of this paper is organized as follows. Section 2 introduces the models used in B-iTRS. Section 3 details the proposed B-iTRS. Section 4 analyzes B-iTRS with mathematical method. Section 5 evaluates our models and algorithms with extensive simulations. Finally, the conclusion is presented in Section 6.

2. System Framework and Models

2.1. Typical WSNs Scenario. This paper considers large multihop WSNs whose nodes are distributed randomly in a two-dimensional space. We assume that (1) each node has a single channel, (2) the interference range R is equal to

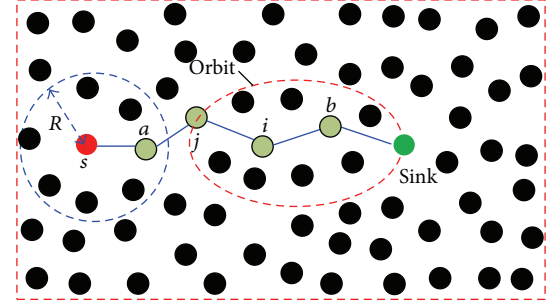


FIGURE 1: An example of MWSNs' topology.

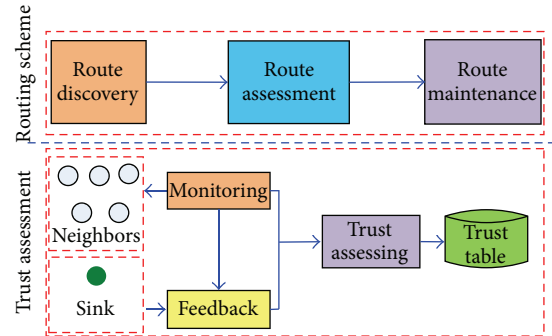


FIGURE 2: B-iTRS framework.

the transmission range, and (3) all communication links are bidirectional.

Based on the assumptions, WSNs can be abstracted as a graph $G = (V, E)$, where V is the set of all nodes and E is the set of all edges. Each edge $e(i, j) \in E$ denotes that the two nodes are located within each other's transmission range.

In some circumstances, the Sink node moves along a certain orbit in the field and broadcasts periodically its current positions to improve the efficiency of collected data. An example of such WSNs' topology is shown in Figure 1.

2.2. The Framework of B-iTRS. Based on the above discussion, Figure 2 illustrates the design of the proposed B-iTRS framework which consists of trust assessment and routing scheme.

For trust assessment, each node monitors its neighbors' behavior in real time and receives the feedback from the Sink node. Based on the neighbors' behavior and feedback information, B-iTRS assesses trusts of neighbors; that is, the trust of corresponding neighbor is added by a different positive, negative, or zero value according to the behavior's weight or Sink's feedback. In most cases, the trust consists of two sections—direct trust and indirect trust. Because of considerable traffic load and computing cost, this paper only considers the direct trust.

For routing scheme, perceptive ACO is used to find and assess routes. Each node s sends perceptive forward-ants to find routes to the Sink, where the cross-layer perception [22] is introduced to support route selection. When an ant hops from node i to node j on the way to source, it firstly senses its own load and the trust with respect to node i .

Then, B-iTRS assesses the trust, load, and availability of route j, i, \dots Sink. If the availability value of the route is less than a specific threshold $TH_{A,Lower}$, B-iTRS kills the backward-ant and discards the route. Otherwise, the route is inserted in the proper position of the route table of node j according to the availability value. The process is repeated until the backward-ant reaches source. When route failures happen, route maintenance is triggered to recover the route rapidly.

2.3. Trust Assessment Model. In B-iTRS, each node monitors its neighbors' behaviors in real time and then analyzes them to get their trust weight, since analyzing node's behavior can show whether the node is selfish, is acting like a black hole, is carrying out a modification or fabrication attack, is inducing latency delays by delaying the retransmission of the packet [23], and so forth. In order to alleviate the calculating burden, we do not adjust the trust weight of behaviors automatically but adopt the prior trust weight which is determined by offline decision.

If a new neighbor j of node i arises, node i endows it with initial trust value $TRUST_{i,j}$, which is determined by

$$TRUST_{i,j} = \begin{cases} \frac{1}{n} \sum_{k=1}^n trust_{i,k}, & \text{if } n \geq 1 \\ trust_c, & \text{if } n = 0, \end{cases} \quad (1)$$

where $trust_{i,k}$ is the trust value of node i with respect to the k th neighbor, n is the number of neighbors, and $trust_c$ is a constant.

Once node i obtains a specific behavior about neighbor j by monitoring it, node i analyzes the behavior based on prior knowledge to identify the behavior of neighbor j and get the trust weight of the behavior and then updates the trust value of node i with respect to neighbor j by

$$trust_{i,j}(t+1) = \begin{cases} TRUST_{i,j} + \text{weight}, & \text{if } t = 1 \\ trust_{i,j}(t) + \text{weight}, & \text{if } t > 1, \end{cases} \quad (2)$$

where weight is the trust weight of the behavior identified, $trust_{i,j}(t)$ is the trust value after the t th updating, and $trust_{i,j}(t+1)$ is the trust value after the $(t+1)$ th updating.

Once node i receives the feedback about its neighbors from the Sink node, node i analyzes feedback information to modify the trusts of its neighbors by

$$trust_{i,j}(\text{new}) = trust_{i,j}(\text{old}) + \text{weight}(j), \quad (3)$$

where $trust_{i,j}(\text{new})$ is the trust value after updating, $trust_{i,j}(\text{old})$ is the trust value before updating, and $\text{weight}(j)$ is the trust weight of node j based on the feedback of Sink node, where $\text{weight}(j)$ is standardized to the same range of $trust_{i,j}$.

2.4. Perceptive ACO Path-Finding Model. We improve the traditional ants to perceptive ants based on ACO to obtain path-finding and path-assessment models by introducing cross-layer perception [22]. For MWSNs expressed by graph $G = (V, E)$, each edge $e(i, j) \in E$ has a variable artificial

pheromone $\tau(i, j)$, which is an indication of usage of the edge and is modified by perceptive ants when they hop from node i to node j .

A perceptive ant located in node i uses $\tau(i, j)$ of node j to compute the probability of node j as next hop:

$$P_{ij} = \frac{\tau(i, j)}{\sum_{j \in N_i} \tau(i, j)}, \quad j \in N_i, \quad (4)$$

where P_{ij} satisfies $\sum_{j \in N_i} P_{ij} = 1$ and N_i is the set of one-hop neighbors of node i .

Once the perceptive ant moves from node i to node j , it senses two metrics L_j and $trust_{i,j}$ to measure the link status, where L_j denotes the length of transmission waiting queue of node j and $trust_{i,j}$ denotes the trust value of node i with respect to the j . If it is the $(t+1)$ th time that perceptive ants pass by the edge $e(i, j)$, depositing pheromone on $e(i, j)$ and evaporating pheromone from $e(k, j)$, $k \neq i$, respectively, are followed by

$$\begin{aligned} \tau_{t+1}(i, j) &= \begin{cases} \tau_t(i, j) + \frac{L_0 \cdot trust_{i,j}}{L_j \cdot T_0} \cdot \Delta\tau, & \text{if } \frac{L_0 \cdot trust_{i,j}}{L_j \cdot T_0} \leq K_0 \\ \tau_t(i, j) + K_0 \cdot \Delta\tau, & \text{otherwise,} \end{cases} \end{aligned} \quad (5)$$

$$\begin{aligned} \tau_{t+1}(k, j) &= \begin{cases} \left(1 - \frac{L_j \cdot T_0}{L_0 \cdot trust_{i,j}}\right) \cdot \tau_t(k, j), & \frac{L_j \cdot T_0}{L_0 \cdot trust_{i,j}} \in [0, 1] \\ (1 - U_0) \cdot \tau_t(k, j), & \text{otherwise,} \end{cases} \end{aligned} \quad (6)$$

where $L_0, T_0, K_0, \Delta\tau$, and U_0 are constants and $U_0 \in (0, 1)$. $\tau_t(i, j)$ denotes the pheromone on edge $e(i, j)$ after the t th depositing or evaporating. Equations (4), (5), and (6) are used to find routes in B-iTRS.

When a backward perceptive ant moves from node i to node j on the way to source, the link quality and link load are assessed with the metrics, respectively, following

$$L.R_{dj} = \begin{cases} L.R_{di} + T_0, & \text{if } trust_{ij} > T_0 \\ L.R_{di} + T_1, & \text{if } trust_{ij} < T_1 \\ L.R_{di} + trust_{ij}, & \text{otherwise,} \end{cases} \quad (7)$$

$$L.L_{dj} = \begin{cases} L.L_{di} + L_0, & \text{if } L_j > L_0 \\ L.L_{di} + L_j, & \text{if } L_j \leq L_0, \end{cases} \quad (8)$$

where T_0, T_1 , and L_0 are constants and $L.R_{dj}$ and $L.L_{dj}$ denote the link reliability and link load from node d to node j , respectively. Equations (7) and (8) are used to assess the discovered routes in B-iTRS.

2.5. *PAO Path-Assessment Model.* In this section, we improve PAO to acquire PAO path-assessment model. From [12–15], the flux through each plasmodial tube follows

$$Q_{ij} = \frac{C_{ij}(P_i - P_j)}{D_{ij}} = \frac{C_{ij} \cdot \Delta P_{ij}}{D_{ij}}, \quad (9)$$

where $\Delta P_{ij} = P_i - P_j$ is the pressure difference of two ends of the tube, C_{ij} is a measure of the conductivity of the tube, and D_{ij} is the length of the tube. Physarum forages for distributed food sources through adapting its plasmodium to change the flux of each tube, which is described by

$$\frac{d}{dt}C_{ij} = \varphi(|Q_{ij}|) - \delta C_{ij}, \quad (10)$$

where δ is a decay rate of the tube and $\varphi(\cdot)$ is a monotonically increasing continuous function satisfying $\varphi(0) = 0$.

Since PAO comes from fluid dynamics and cannot be directly used in MANETs, we replace C_{ij} with the link trust trust_{ij} , D_{ij} with H_{ij} which denotes the hops from node i to node j , and ΔP_{ij} with $\Delta L_{ij} = L_i - L_j$. Thus, we obtain

$$Q_{ij} = \frac{C_{ij} \cdot \Delta P_{ij}}{D_{ij}} = \frac{\text{trust}_{ij} \cdot \Delta L_{ij}}{H_{ij}}, \quad (11)$$

$$\begin{aligned} \frac{d}{dt}\Delta\text{trust}_{ij} &= \varphi(|Q|) - \delta\Delta\text{trust}_{ij} \\ &= \left(\frac{\text{trust}_{ij} \cdot \Delta L_{ij}}{H_{ij}} \right)^\mu - \delta\Delta\text{trust}_{ij}, \end{aligned} \quad (12)$$

where Q_{ij} is the virtual flux of communication through the wireless link $e(i, j)$, δ is the rate of trust changing, and μ is a constant satisfying $\mu > 0$. Equation (12) is used to select the optimal route in B-iTRS.

3. B-iTRS Protocol

The B-iTRS consists of trust assessment and routing scheme. The trust assessment is running in each node independently to acquire its neighbors' trusts. The route scheme works based on PACO and PAO. The module structure is shown in Figure 3.

3.1. *Data Structures in B-iTRS.* There are mainly three kinds of data structures in B-iTRS—perceptive ant structure, route table, and pheromone table. The structure of perceptive ant is seven-tuple $\langle \text{Source}, \text{Sequence_No}, \text{Type}, \text{Hops}, \text{Path}, \text{Link_Reliability}, \text{Link_Load} \rangle$. Source field stores the source node address. Sequence_No field stores the sequence number tagged to each ant. Source and destination nodes incrementally generate a Sequence_No each time forward- or backward-ants are sent out. The pair $\langle \text{Source}, \text{Sequence_No} \rangle$ can uniquely identify the ants' generation. Type field indicates the ants' type. There are four types of ants: the first is perceptive forward-ant used for finding routes; the second is perceptive backward-ant used for returning routes to source; the third is notification ant

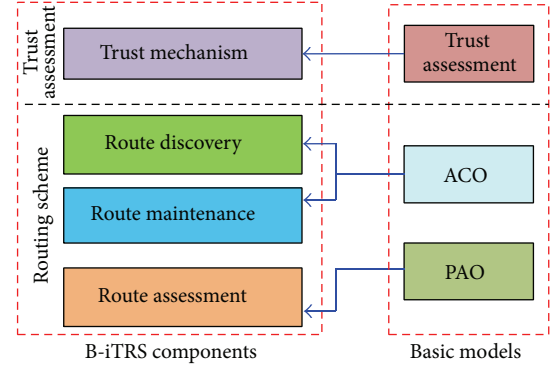


FIGURE 3: B-iTRS module structure.

used for sending notification to other nodes; and the last is error ant used for sending error to other nodes. For forward-ant, the Hops field indicates the maximum hops that one ant can move. For backward-ant, the Hops field stores the hops of a route. Path field stores the sequence of nodes from source to destination. Link_Reliability and Link_Load measure the link reliability and link load of the discovered routes, respectively.

Each node needs to maintain a route table whose structure is six-tuple $\langle \text{Source}, \text{Hops}, \text{Link_Reliability}, \text{Link_Load}, \text{Path}, \text{Sequence_No} \rangle$. Hops field stores the hops from source to destination. The structure of pheromone table is triple $\langle i, j, \text{Value} \rangle$ which expresses the pheromone of link $e(i, j)$ is Value.

3.2. *Route Discovery.* When a source s wishes to send data packet to Sink, it looks up its route table firstly. If only one route is found, the route discovery process is over. If multiple routes are found, the route selection is performed according to (12).

If there is no route from source s to Sink in route table, the route discovery will be triggered. The node s sets a timeout T_0 and sends perceptive forward-ants to find new routes, where Hops fields are set to the allowable maximum value, which means each perceptive forward-ant will die only to find Sink, a node with a route to Sink, or move limited maximum hops.

When a forward-ant reaches an intermediate node, it checks whether unexpired routes to destination Sink already exist in the route table firstly. If there are one or more unexpired routes, the optimal route to Sink is selected according to (9), and a corresponding perceptive backward-ant is generated and sent back to source along the discovered path. Otherwise, the transition probability, pheromone depositing, and evaporating are calculated following (4), (5), and (6), respectively. Once a perceptive forward-ant reaches Sink, a corresponding backward-ant is generated and sent back to source along the discovered path.

When the perceptive backward-ant reaches each intermediate node i on the way to source, B-iTRS will perform two operations. Firstly, the pheromone depositing, pheromone evaporating, link quality, and link load are calculated following (5), (6), (7), and (8), respectively. Secondly, the new route will be added to the route table of node i . After the perceptive

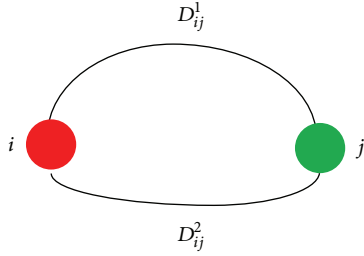


FIGURE 4: B-iTRS module structure.

backward-ant reaches source s , the new route will be added to route table.

If T_0 is time out and no route is found, the route discovery fails. If only one route is found, the route discovery process is over. If multiple routes are found, (9) is used to determine the optimal route.

3.3. Route Maintenance. Route maintenance handles routing failures especially caused by node mobility or breakdown which is very common in MWSNs. If the route failures happen, the upstream node of the broken link will trigger a repair procedure to find new routes to Sink, which is similar to the route discovery. If an alternative route is found, the transmission is going on and a notification ant is sent to source to update route table of each node on the path. If no route is found, an error ant will be sent to source as well as updating route table of each node on the path. After receiving an error ant, if the source still needs to transmit the data packet, it can select the backup route or other unexpired routes found in route table and even initiate a new procedure of route discovery among zones.

4. B-iTRS Analysis

In this section, we analyze the feasibility of B-iTRS by mathematical theoretical analysis. We study the cases in which two nodes connected to the same node compete to be the next hop, as shown in Figure 4.

There are two nodes i and j_1 . For simplicity, we hereafter replace D_{ij_1} , D_{ij_2} , Q_{ij_1} , Q_{ij_2} , ΔP_{ij_1} , and ΔP_{ij_2} by D_1 , D_2 , Q_1 , Q_2 , ΔP_1 , and ΔP_2 , respectively. From (9) and (10), the virtual fluxes along each path are calculated as

$$\begin{aligned} Q_1 &= \frac{\Delta P_1/D_1}{\Delta P_1/D_1 + \Delta P_2/D_2}, \\ Q_2 &= \frac{\Delta P_2/D_2}{\Delta P_1/D_1 + \Delta P_2/D_2}. \end{aligned} \quad (13)$$

Since Q_1 and Q_2 are nonnegative, adaptation equation (9) becomes

$$\begin{aligned} \frac{d}{dt}(\Delta P_1) &= \varphi(Q_1) - \delta \cdot \Delta P_1, \\ \frac{d}{dt}(\Delta P_2) &= \varphi(Q_2) - \delta \cdot \Delta P_2. \end{aligned} \quad (14)$$

Setting $\varphi(Q) = Q^\mu$, $(d/dt)(\Delta P_1) = 0$, and $(d/dt)(\Delta P_2) = 0$, we have

$$\begin{aligned} \left(\frac{\Delta P_1/D_1}{\Delta P_1/D_1 + \Delta P_2/D_2} \right)^\mu &= \delta \cdot \Delta P_1, \\ \left(\frac{\Delta P_2/D_2}{\Delta P_1/D_1 + \Delta P_2/D_2} \right)^\mu &= \delta \cdot \Delta P_2. \end{aligned} \quad (15)$$

After some calculations, we obtain

$$\begin{aligned} \Delta P_1 &= \frac{1}{\delta} \left[\frac{1}{\left(1 + (D_1/D_2)^{1/1-\mu}\right)} \right]^\mu, \\ \Delta P_2 &= \frac{1}{\delta} \left[\frac{1}{\left(1 + (D_2/D_1)^{1/1-\mu}\right)} \right]^\mu. \end{aligned} \quad (16)$$

From (10), if we suppose ΔL_{ij} is constant and use trust_{ij} replacing C_{ij} , H_{ij} replacing D_{ij} , and ΔL_{ij} replacing D_{ij} , we obtain

$$\begin{aligned} \Delta \text{trust}_1 &= \frac{1}{\delta} \left[\frac{1}{\left(1 + (H_1/H_2)^{1/1-\mu}\right)} \right]^\mu, \\ \Delta \text{trust}_2 &= \frac{1}{\delta} \left[\frac{1}{\left(1 + (H_2/H_1)^{1/1-\mu}\right)} \right]^\mu. \end{aligned} \quad (17)$$

Namely, if we suppose ΔL_{ij} is constant, there is an equilibrium point given by $(\Delta \text{trust}_1, \Delta \text{trust}_2)$. If we suppose trust_{ij} is constant, there is an equilibrium point given by $(\Delta L_1, \Delta L_2)$. If both ΔL_{ij} and trust_{ij} are not constants, there is a more complicated equilibrium point. Therefore, the B-iTRS is always convergent, which is very important for a routing protocol.

5. Simulation Results and Analysis

We analyze B-iTRS in Network Simulator ns-2 (version 2.34) and compare its simulation results with those of AODV, AntHocNet, and TGRP. In the base simulation scenario, the Sink node is placed in the center of a rectangular area of 600 m \times 600 m, and 100 nodes are uniformly placed in the area and move according to the random way mobility model (RWP) [24]. The Sink moves along an ellipse orbit in the center of the rectangular area. It broadcasts periodically its current positions, as shown in Figure 1. In the model, each node moves towards a random direction at a speed uniformly distributed [0, 10 m/s]. Once a node reaches a target position, it pauses for resting 2 s to send or transmit data packet and then moves forward in the same way. The data traffic is generated by 20 constant bit rate (CBR) sources with sending rates of single 64 bytes every 2 s. The radio propagation range and data rate are set to 50 m and 2 Mbit/s, respectively. The parameters are shown in Table 1.

The simulation is run for 600 s each time. We run each simulation scenario 10 times to acquire the average values of

TABLE I: Simulation parameters.

Parameters	Value
Simulation range	600 m × 600 m
Node's number	100
Max speed	10 m/s
Min speed	0 m/s
Propagation range	50 m
Data rate	2 Mbit/s
Node moving time	3 s
Node rest time	2 s
Simulation time	600 s

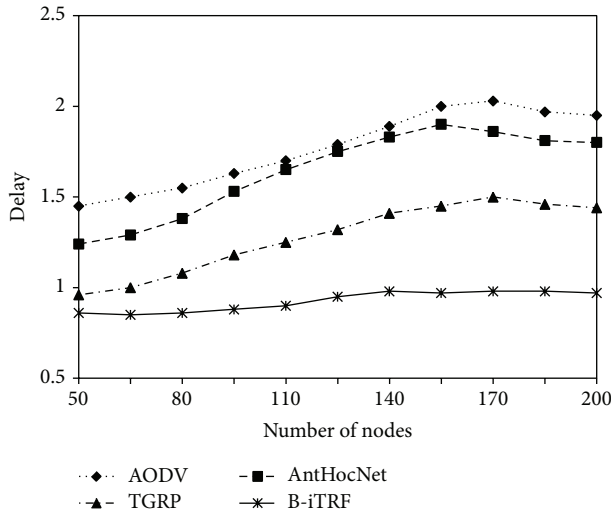


FIGURE 5: Delay versus number of nodes.

results and compare them. The results and analysis are shown as follows.

Figure 5 shows end-to-end delay versus number of nodes when the ratio of malicious nodes is 8%. Since AODV is on-demand protocol, its end-to-end delay is the worst of the four protocols. Because AntHocNet can proactively test existing paths and explore new ones during the course of communication session, its end-to-end delay is better than that of AODV. Since both B-iTRF and TGRP can deal with malicious nodes, their end-to-end delays are much better than those of AODV and AntHocNet.

Figure 6 shows end-to-end delivery ratio versus number of nodes when the ratio of malicious nodes is 8%. In the first stage, each delivery ratio increases rapidly with the increase of the number of nodes. After the number of nodes reaches a specific value, the delivery ratio will keep a stable value approximately. Since B-iTRF and TGRP adopt security mechanism, their delivery ratios are similar and greater than those of AODV and AntHocNet.

Figure 7 shows overhead versus number of nodes when the ratio of malicious nodes is 8%. Since AODV is a purely reactive and AntHocNet is hybrid, their control overheads are the least and the second least in the four protocols, respectively. However, their control overheads increase rapidly with

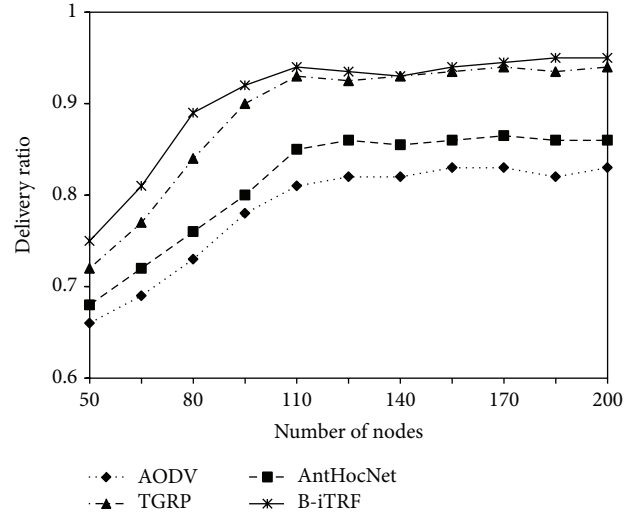


FIGURE 6: Delivery ratio versus number of nodes.

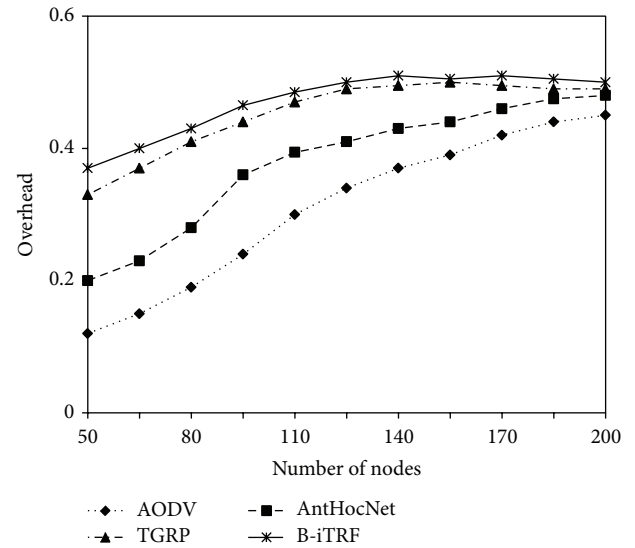


FIGURE 7: Overhead versus number of nodes.

the increase of the number of nodes. Since B-iTRF needs to deal with neighbors' trusts and Sink feedback, its control overhead is greater than that of TGRP which only needs to deal with trust.

Figure 8 shows end-to-end delay versus ratio of malicious nodes. Since AODV and AntHocNet cannot deal with attacks from malicious nodes, their end-to-end delay increases with the increase of malicious nodes, and the increment is larger and larger. Since both B-iTRF and TGRP can deal with malicious nodes, their end-to-end delays are similar.

Figure 9 shows delivery ratio versus ratio of malicious nodes. Each delivery ratio decreases gradually with the increase of the ratio of malicious nodes, and the decrement is greater and greater. In the first stage, each delivery ratio decreases little with the increase of the ratio of malicious nodes. After the ratio of malicious nodes reaches a specific

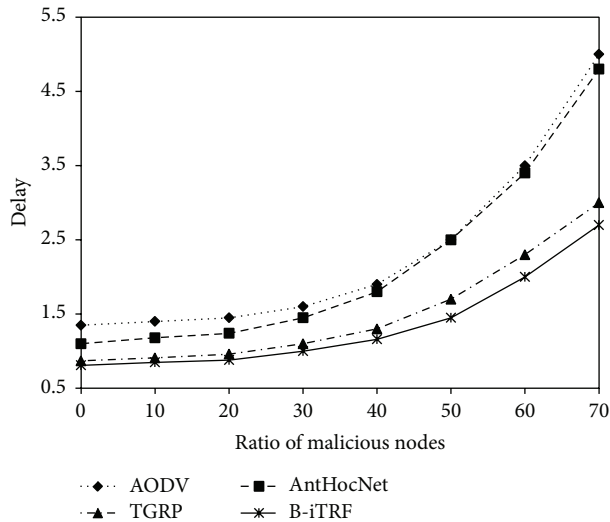


FIGURE 8: Delay versus ratio of malicious nodes.

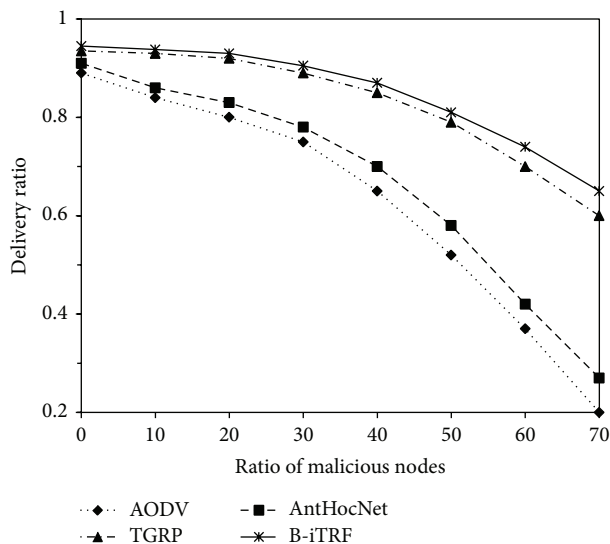


FIGURE 9: Delivery ratio versus ratio of malicious nodes.

value, the two delivery ratios will decrease rapidly. Since both B-iTRF and TGRP adopt security mechanism, their delivery ratios are similar and decrease slowly.

6. Conclusion and Future Work

In this paper, we present B-iTRF, a trusted routing scheme based on bio-inspired method. B-iTRF uses perceptive ants to reactively maintain the route table, where the link status metrics are sensed by perceptive ants to support path finding. Then, B-iTRF uses PAO to select the optimal route from multiple routes. In fact, B-iTRF is devoted to combining the advantages of both ACO and PAO to improve the effective performance, which is verified by simulation results. The proposed scheme can be used in both WSNs and MANETs scenario. In future work, we consider extending link status

metrics (e.g., interference, energy) and introducing actual mobility model of nodes into B-iTRF to make it fit in with real WSNs environment.

Conflict of Interests

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

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC) under Grant no. U1404611, no. U1204614, and no. 61370221 and by the key project of the Education Department of Henan Province under Grant no. 14B520031, in part by Program for Science & Technology Innovative Research Team in University of Henan Province under Grant no. 14IRTSTHN021, and in part by the Program for Science & Technology Innovation Talents in the University of Henan Province under Grant no. 14HASTIT045.

References

- [1] G. Zhan, W. Shi, and J. Deng, "Design and implementation of TARE: a trust-aware routing framework for WSNs," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 2, pp. 184–197, 2012.
- [2] X. Yin and J. Yang, "Shortest paths based web service selection in internet of things," *Journal of Sensors*, vol. 2014, Article ID 958350, 10 pages, 2014.
- [3] W. Guo, R.-Z. Xu, and B. Liu, "Research on subjective trust routing algorithm for mobile ad hoc networks," in *Proceedings of the 6th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '10)*, September 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] D. Dasgupta, S. Yu, and F. Nino, "Recent advances in artificial immune systems: models and applications," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1574–1587, 2011.
- [6] J. Rao and A. O. Fapojuwo, "A battery aware distributed clustering and routing protocol for wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '12)*, pp. 1538–1543, April 2012.
- [7] O. Yang and W. Heinzelman, "Sleeping multipath routing: a trade-off between reliability and lifetime in wireless sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–5, Houston, Tex, USA, December 2011.
- [8] G. Trajcevski, F. Zhou, R. Tamassia, B. Avci, P. Scheuermann, and A. Khokhar, "Bypassing holes in sensor networks: load-balance vs. latency," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '11)*, December 2011.
- [9] J. Chen, Y. Chen, L. Zhou, and Y. Du, "A data gathering approach for wireless sensor network with quadrotor-based mobile sink node," *KSII Transactions on Internet and Information Systems*, vol. 6, no. 10, pp. 2529–2547, 2012.

- [10] M. Gunes, U. Sorges, and I. Bouazizi, "ARA—the ant-colony based routing algorithm for MANETs," in *Proceedings of the International Conference on Parallel Processing Workshops*, pp. 79–85, August 2002.
- [11] G. Di Caro, F. Ducatelle, and L. M. Gambardella, "AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, 2005.
- [12] A. Tero, R. Kobayashi, and T. Nakagaki, "A mathematical model for adaptive transport network in path finding by true slime mold," *Journal of Theoretical Biology*, vol. 244, no. 4, pp. 553–564, 2007.
- [13] K. Li, C. E. Torres, K. Thomas, L. F. Rossi, and C.-C. Shen, "Slime mold inspired routing protocols for wireless sensor networks," *Swarm Intelligence*, vol. 5, no. 3-4, pp. 183–223, 2011.
- [14] M. Zhang, C. Xu, J. Guan, R. Zheng, Q. Wu, and H. Zhang, "P-iRP: physarum-inspired routing protocol for wireless sensor networks," in *Proceedings of the IEEE 78th Vehicular Technology Conference (VTC '13)*, September 2013.
- [15] M. Zhang, C. Xu, J. Guan, R. Zheng, Q. Wu, and H. Zhang, "A novel Physarum-inspired routing protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 483581, 12 pages, 2013.
- [16] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 90–100, IEEE, New Orleans, La, USA, February 1999.
- [17] Y. Ren, V. I. Zadorozhny, V. A. Oleshchuk, and F. Y. Li, "A novel approach to trust management in unattended wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 7, pp. 1409–1423, 2014.
- [18] I.-R. Chen, F. Bao, M. Chang, and J.-H. Cho, "Dynamic trust management for delay tolerant networks and its application to secure routing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 5, pp. 1200–1210, 2014.
- [19] J. S. Priyanka, S. Tephillah, and A. Balamurugan, "Malicious node detection using minimal event cycle computation method in wireless sensor networks," in *Proceedings of the International Conference on Communications and Signal Processing (ICCSP '14)*, pp. 905–909, April 2014.
- [20] S. Indhu Lekha and R. Kathioli, "Trust based certificate revocation of malicious nodes in MANET," in *Proceedings of the IEEE International Conference on Advanced Communication Control and Computing Technologies*, pp. 1185–1189, May 2014.
- [21] X. O. Wang, W. Cheng, P. Mohapatra, and T. Abdelzaher, "Enabling reputation and trust in privacy-preserving mobile sensing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 12, pp. 2777–2790, 2014.
- [22] Y. Cao, C. Xu, J. Guan, J. Zhao, and H. Zhang, "Cross-layer cognitive CMT for efficient multimedia distribution over multi-homed wireless networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '13)*, pp. 4522–4527, April 2013.
- [23] L. Abusalah, A. Khokhar, and M. Guizani, "A survey of secure mobile Ad hoc routing protocols," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 78–93, 2008.
- [24] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996.

Research Article

A Novel Digital Certificate Based Remote Data Access Control Scheme in WSN

Wei Liang,¹ Zhiqiang Ruan,² Hongbo Zhou,¹ and Yong Xie¹

¹Department of Software and Engineering, Xiamen University of Technology, Xiamen, Fujian 361024, China

²Department of Computer Science, Minjiang University, Fuzhou, Fujian 350108, China

Correspondence should be addressed to Zhiqiang Ruan; rzq_911@163.com

Received 12 November 2014; Accepted 17 April 2015

Academic Editor: Fei Yu

Copyright © 2015 Wei Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A digital certificate based remote data access control scheme is proposed for safe authentication of accessor in wireless sensor network (WSN). The scheme is founded on the access control scheme on the basis of characteristic expression (named CEB scheme). Data is divided by characteristics and the key for encryption is related to characteristic expression. Only the key matching with characteristic expression can decrypt the data. Meanwhile, three distributed certificate detection methods are designed to prevent the certificate from being misappropriated by hostile anonymous users. When a user starts query, the key access control method can judge whether the query is valid. In this case, the scheme can achieve public certificate of users and effectively protect query privacy as well. The security analysis and experiments show that the proposed scheme is superior in communication overhead, storage overhead, and detection probability.

1. Introduction

The WSN is a dynamic wireless network formed by multiple microsensor nodes. It can be used for continuous environment monitoring. The nodes have the features of low consumption and low cost. Furthermore, they can realize data collection, data interaction, data transmission, and distributed cooperation [1–3]. However, WSN has some shortages, such as intrinsic vulnerability, limit sensor nodes, random deployed nodes, dynamic change of network topology, and unstable wireless channel. Furthermore, since WSN is always deployed in rugged environment, uninhabited area, or enemy positions, some unique security risks exist, such as intercept, leakage, denial of service, false injection, tampering, and replay attacks [4, 5].

In this case, it is an important issue to create a safe and reliable working scene for WSN, which relates to practicability and promotion of WSN. The features of sensor nodes are seriously limited in terms of calculating speed, power supply energy, communication ability, and storage space. Consequently, it is necessary to design an effective security mechanism under the limit conditions. In this way, the data stored in sensor nodes will be complete, confidential, and reliable and have the ability against intercept and capture

in data transmission. The unauthorized query and sensitive information leakage of users are prevented.

Recently, data collection in sensor network needs users to pay for the access. Meanwhile, the privacy of data access is an inevitable problem. For instance, some users are not willing to leak the information about the time of accessing, interesting data type, and retrieved nodes. The identities are expected to be unknown to network owner and other users. In this way, they can protect their benefits from being damaged by potential competitors. Therefore, it is important to realize public user authentication while satisfying user anonymous requirement. So far, anonymous authentication is widely concerned but only on authentication problem. If the identity verification is successful, the nodes will provide data for user without considering anonymity. SPYC [6] is the first anonymous access control scheme by collecting data through one or several base stations. He et al. [7] present a distributed access control with privacy support in wireless sensor networks. All of users are divided into groups. Each group has various grades. In user query, the request is sent with group identity. It can authenticate user's validity and also protect privacy of user's identity. However, this way reveals user's privacy in dividing process. Since the number of groups is

limited, user's identity and his interesting data can be deduced by network provider with exhaustive analysis method. It is not a real distributed algorithm. Bethencourt et al. [8] propose a ciphertext policy attribute based encryption, called BSW scheme. Secret sharing method is employed in encryption for strict access control. The private key is related with characteristic set in BSW. An access structure is implicated in ciphertext. If the characteristic of private key satisfies the access structure, the private key can be used for decryption. In BSW, polynomial interpolation is required to reconstruct the key. Therefore, many operations of complex matching and exponentiation should be performed in decryption. Wang and Li [9] present an authorization method based on access control list (ACL). User will obtain ACL and certificate in advance. (n, t) threshold method is used in authentication. The signature employs the asymmetric encryption. If user obtains authentication signatures from t ($t < n$) nodes, the query request will be transmitted. The node with query data will verify and respond to the request. However, the scheme is low efficient and without expandability. Long distance data transmission faces various potential attacks. Cheung and Newport [10] replace secret sharing by random elements in encryption for strict access control. The scheme is named CN. There are two shortages. On one hand, it only supports simple logic combination strategy with low descriptiveness. On the other hand, the sizes of ciphertext and key grow linearly with the increase of the number of characteristics, which degrades efficiency. Ruan et al. [11] present a group-based anonymous scheme to conceal the message transmitted between source node and target node. They further proposed a proxy signature scheme to protect the access of data [12]. However, they focus on the privacy of data but not the privacy of users, which will be solved in this paper.

In our work, each user should get certificate before data collection, and the certificate should ask for or buy from network provider. User can access data with the certificate in the network. The sensor node will verify validity of the certificate and then provide the requested data to user. Each sensor node can verify validity of the certificate, but user's identity is unknown to all entities. In our scheme, the network provider can prevent unauthorized user access and protect user privacy. The generation of certificate is based on blind signature [13]. In traditional digital signature schemes, signer knows all information about his signature. But signer cannot get what he signed in blind signature system. The proxy blind signature is generated on proxy signature and blind signature, which is widely applied in payment of electronic currency and greatly protects user's benefits. Meanwhile, the proxy blind signature is suitable for the proposed application scenarios. However, anonymity of blind signature may be utilized by hostile users for unlimited access. Consequently, a novel safe access control algorithm based on digital certificate is proposed from the view of security requirement. It can address the security issues in access control.

2. Network Model and Hypothesis

As shown in Figure 1, suppose that there are a network service provider P , an intermediate proxy A , and some

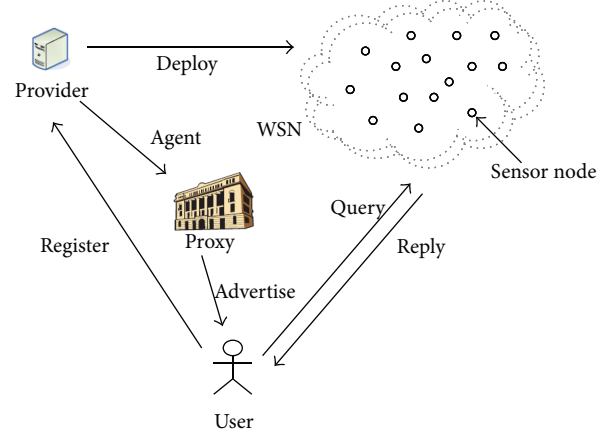


FIGURE 1: Distributed network access structure.

users U in WSN. N nodes in WSN continuously monitor target environment and provide interesting data to users. No reliable base station in network will connect intranet with outside network. The collected data will be restored in local nodes or other nodes. Therefore, user can obtain data directly from nodes. The nodes are supposed to know their geographical location information by existing locating algorithm.

We suppose users can conspire, forge a certificate, and even capture part of nodes to obtain their interesting information. Users are not willing to reveal their identities and the way to access data but want to obtain information about other users as much as possible. Of course, they will have a sharp practice if it is profitable. Different from that of general wireless sensor network, users will not sponsor denial of service (DoS) since it is no good for data acquisition. Users will not escape access control and collect data by directly capturing many nodes since enormous costs and efforts are required. On the contrary, users will capture a small part of nodes for reuse of certificate.

3. User Authentication and Privacy Protection

3.1. Scheme Description. User access control includes three stages: proxy stage, certificate generation, and verification. The symbol description is stated as follows:

p, q : two big prime numbers satisfying $q \mid p-1$, q and $p-1$ being relatively prime,

g : an element with rank q in Z_p^* , named generator,

m : message needed to be signed,

m_w : a valid evidence of authorized intermediary proxy, including identity information about network provider, intermediary proxy, type of signed message, authorized scope, and valid date,

$x_P, x_A \in Z_p^*$: respective private keys of network provider and intermediary proxy,

$y_P = g^{x_P} \pmod p$, $y_A = g^{x_A} \pmod p$: respective public keys of network provider and intermediary proxy,

$H(\cdot)$, $h(\cdot)$: hash function with high security in cryptography,

\parallel : connection string.

3.1.1. *Proxy Stage*. Four steps are included in this stage:

- (1) U sent registration information (without user's identity) to P ,
- (2) P randomly selects data $k \in_R Z_p^*$ and calculates $K = g^k \pmod p$, $s = x_P + k \cdot H(m_w \parallel K) \pmod q$,
- (3) P transmits (K, s) and m_w to proxy A in a safe channel,
- (4) A receives (K, s) and verifies if $g^s = y_P K^{H(m_w \parallel K)} \pmod p$ is satisfied; if done, A accepts proxy task and further calculates proxy key with $s' = s + x_A$.

3.1.2. *Certificate Generation*. In this stage, the intermediary proxy A needs to do the following work.

- (1) A randomly selects data $\lambda \in_R Z_p^*$ and calculates $t = g^{\lambda + x_A} \pmod p$.
- (2) A sends (K, t) to user U (user with certificate can access data on nodes).
- (3) U receives the information and randomly selects data $a, b \in Z_p^*$ and calculates the following values:

$$\begin{aligned} \mu &= t^a (y_P y_A K^{H(m_w \parallel K)})^{ab} \pmod p, \\ e &= h(m \parallel \mu) \pmod q, \\ e' &= a^{-1} e + b \pmod q. \end{aligned} \quad (1)$$

If $u = 0$, repeat step (3) until $u \neq 0$. After that, e' is sent to A .

- (4) A receives e' and computes $s'' = e' s' + \lambda + x_A$ as a signature of message. s'' is sent to U .
- (5) U receives s'' and computes $\varphi = g^{s'' a} \pmod p$. (m, m_w, φ, e, K) is regarded as proxy blind signature of message m , called certificate.

3.1.3. *Verification*. Each sensor node has y_P and y_A before deployment. Network provider can dynamically update y_P and y_A by using method in [14]. Once the certificate is obtained by user, U can access the WSN and collect data from nodes. Any node N_i that receives certificate (m, m_w, φ, e, K) will verify whether the equation $e = h(m \parallel \mu) \pmod q$ is satisfied. According to $e = h(m \parallel \mu) \pmod q$, we only need to prove whether $\mu =$

$\varphi (y_P y_A K^{H(m_w \parallel K)})^{-e} \pmod q$ is satisfied. We have the following deduction process:

$$\begin{aligned} \varphi (y_P y_A K^{H(m_w \parallel K)})^{-e} &= g^{s'' a} (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= (g^{e' s' + \lambda + x_A})^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= g^{(a^{-1} e + b) s' a} \cdot t^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= g^{e s'} \cdot g^{s' a b} \cdot t^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= (g^{s + x_A})^e \cdot (g^{s + x_B})^{a b} \cdot t^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= (g^{x_P + k H(m_w \parallel K)} g^{x_A})^e \\ &\quad \cdot (g^{x_P + k H(m_w \parallel K)} g^{x_A})^{a b} \cdot t^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= (y_P K^{H(m_w \parallel K)} y_A)^e \\ &\quad \cdot (y_P K^{H(m_w \parallel K)} y_A)^{a b} \cdot t^a \\ &\quad \cdot (y_P y_A K^{H(m_w \parallel K)})^{-e} \\ &= t^a \cdot (y_P y_A K^{H(m_w \parallel K)})^{a b} = \mu. \end{aligned} \quad (2)$$

The above expression proves the validity of certificate (m, m_w, φ, e, K) . After that, N_i will check again. Only the two steps are successful: node N_i will provide data to user U according to the certificate.

3.2. *Certificate Detection Algorithm*. Each certificate (m, m_w, φ, e, K) consists of simple characters or digital number, which are unable to track. Hostile users may reuse their certificates and may not fear being caught. Therefore, node should verify whether the certificate is used before responding, called certificate detection. It performs before using certificate and after signature verification. The witness node is introduced for verifying abused certificates effectively.

Suppose the certificate is successfully used by user U for $a-1$, $a \geq 1$ times. Now, U attempts to use the uncompromised node N_i at a th time. The certificate (m, m_w, φ, e, K) is authenticated by the node N_i and should be verified if it is used. P represents the probability of certificate (m, m_w, φ, e, K) being abused in a th use (assume it is successfully used for $a-1$ times). C denotes the communication cost for each data

transmission. M is the storage cost in a th use. N, r, c , and h are, respectively, nodes number, communication range, number of compromised nodes, and the average hops number between two nodes. In this section, three methods are presented to verify the use of certificate.

(1) *Geography Mapping*. Geography mapping (GM) is used in the first method. The way for selecting storage nodes is referenced. We randomly select a number of nodes as witness nodes. GPSR [13] is used to find witness nodes. Node N_i receives certificate (m, m_w, φ, e, K) ; after that, we randomly select b witness nodes with GPSR for certificate detection. The positions of witness nodes are calculated by $H(m, x) = \{l_i\}_{i=1}^b$. Here, H is the hash function and x is the random number. Node N_i will send m to b witness nodes and set the longest round-trip time of message. Each witness node w_i will judge if m is stored after receiving m . If not, certificate (m, m_w, φ, e, K) will be stored in local memory. Otherwise, w_i will respond to a passive message to node N_i .

Now, we will evaluate detection probability of GM method. b witness nodes are required to verify certificate (m, m_w, φ, e, K) . x_i denotes the generated random number for determining witness position at i th time. One node may be selected in many times of selecting witness nodes. Assume the generated random number x_i in each time is independent. After the certificate being used for $a - 1$ times, the number of witness nodes is $N_w(a - 1)$. The probability of one node not being selected in $a - 1$ times is denoted by $(1 - b/N)^{a-1}$. In this case, the probability for one node being selected for one time at least will be $1 - (1 - b/N)^{a-1}$. Consequently, we have $N_w(a - 1) = N(1 - (1 - b/N)^{a-1})$. If $b/N \ll 1$ is satisfied, we have $N_w(a - 1) \approx N(1 - (1 - b(a - 1)/N))$.

The nodes that are selected as witness nodes are unknown to user U . He can only capture some nodes randomly. Assume there are c nodes being compromised and $c < N$. The probability for witness node being compromised is $c(1 - (1 - b/N)^{a-1}) \approx (a - 1)bc/N$. $b(a - 1)(1 - c/N)$ nodes are not compromised. If none of them is being selected as witness nodes, a th certificate detection fails and the probability is $(1 - b/N)^{b(a-1)(1-c/N)}$.

Consequently, in GS method, the probability for verifying a th certificate abuse is

$$p = 1 - \left(1 - \frac{b}{N}\right)^{(a-1)b(1-c/N)} = \frac{b^2(a-1)(N-c)}{N^2}, \quad (3)$$

$(a \geq 2)$.

a th detection requires $C = (b + \omega)h$ message transmissions. Here, ω is the number of uncompromised nodes that send passive message to node N_i . If $(a - 1)b$ witness nodes are uncompromised, the probability for each node responding to a passive message is b/N . In this case, there are totally $\omega = (a - 1)b^2/N$ passive messages. We have $C = (1 + (a - 1)b/N)bh$. Furthermore, the storage cost is $M \approx ab$.

(2) *Path Feedback*. The second method employs path feedback (PF). It is founded on GM and realized by using the broadcast feature of wireless signal. In the procedure of

sending certificate (m, m_w, φ, e, K) for detection request, all nodes within communication radius of transmission path can receive request message. If one node finds m is stored by itself, it will send a passive message to source node. With the same number of b , PF can greatly improve detection probability of certificate. For example, assume that node w_i is one of b witness nodes selected by node N_i at a th time and it is not selected in $a - 1$ times. In this case, w_i will record m . V is supposed to be a node at path, which acts as witness of certificate and can receive the detection request from node N_i to w_i . Different from GM method, PF allows node V at path to send a passive message to node N_i .

Now, we calculate detection probability of PF method. The hops number between arbitrary two nodes is h . For simplification, we assume N nodes are randomly deployed within the area S and there are $h + 1$ nodes on each detection path from node N_i . The area of circular is $S_r = \pi r^2$. The intersect area of two adjacent circulars is $S' = (4\pi - 3\sqrt{3})r^2/6$. The area formed by H hops path is $S_h = hS_r - (h - 1)S'$. There are b witness nodes, so the number of request messages is b . Therefore, the total area formed by b paths can be calculated by

$$S_b = bS_h - (b - 1)S_r$$

$$= \frac{6\pi r^2 + (2(h - 1)\pi + 3\sqrt{3}(h - 1))br^2}{6}. \quad (4)$$

Similarly, there are c nodes being captured, $c < N$, and the probability is $c(1 - (1 - b/N)^{a-1}) \approx (a - 1)bc/N$. The remaining $b(a - 1)(1 - c/N)$ witness nodes are uncompromised. If none of them has received the request message, a th detection is failure and the probability is $(1 - S_b/S)^{b(a-1)(1-c/N)}$. Consequently, we have

$$p = 1 - \left(1 - \frac{S_b}{S}\right)^{(a-1)b(1-c/N)}. \quad (5)$$

Similar to that of GM method, the communication cost of PF is $C = (b + \omega)h$, $a \geq 1$. If none of $(a - 1)b$ witness nodes are captured, one passive message will always respond with the probability of S_b/S . That is, the number of passive messages is $\omega = (a - 1)S_b b/S$. So, we have $C = (1 + (a - 1)S_b b/S)bh$. In addition, the storage cost is the same as that of GM, $M \approx ab$. Obviously, PF has higher detection probability and lower communication cost and storage cost by comparing to GM:

$$p = 1 - \left(1 - \frac{S_b}{S}\right)^{(a-1)b(1-c/N)}. \quad (6)$$

(3) *Crossline*. The third method is based on crossline theory, called CL method. GM and PF have a common characteristic of compromise among detection probability, communication cost, and storage cost. More witness nodes will improve detection probability but also cause large communication cost and storage cost and vice versa. However, it is different in CL method. CL method is based on crossline technology [15].

Data storage is along with one direction, called “copy path,” but not in one node or several isolated nodes. User query towards another direction is called “query path.” If two paths are crossed, user can query expected data.

In CL method, certificate is regarded as a unique data type and message to be copied or queried. If certificate is received, each node will send a detection request along with any fixed vertical path. If the request is intersected with uncompromised witness nodes which have certificate record, the witness nodes will respond with passive message to source node. Otherwise, the certificate will be regarded as new. The source node will select any horizontal path and copy the certificate to all nodes on the path for storage.

Node N_i randomly generates a position $H(m, x_1)$ after receiving certificate (m, m_w, φ, e, K) . x_1 is arbitrary random number. A proxy query request message with m will be sent to the node at $H(m, x_1)$ by using GPSR. The node, which receives the proxy query request and is near to $H(m, x_1)$, is called proxy query node of N_i , denoted by U_1 . If m is stored in node U_1 , U_1 will send a passive message to node N_i . Otherwise, U_1 will send the query request messages, respectively, along the horizontal and vertical directions. If node N_i receives an abused passive message before the timer expired, the use of certificate (m, m_w, φ, e, K) is refused. Otherwise, the certificate (m, m_w, φ, e, K) is unused. Node N_i will generate a random number x_2 (different with x_1) and send a copied proxy query request message including m to nodes nearby $H(m, x_2)$. The node U_2 , which is closest to $H(m, x_2)$, is regarded as the copied proxy node of N_i after receiving the request message. Then, U_2 stores and sends two copy paths towards horizontal direction. All nodes on copy path will store m .

Now, we analyze the detection probability of CL method. The certificate (m, m_w, φ, e, K) is used for $a - 1$ times. Therefore, there are $a - 1$ copy paths randomly generated in network. At least one node can receive request message for detection. We assume only one node receives the request message. There are $a - 1$ cross nodes. Since the query path of node N_i is unpredictable, user U attempts to use certificate (m, m_w, φ, e, K) at a th time by randomly capturing a number of nodes. If c nodes are compromised by U , the probability of each cross node being captured is c/N . If all of the cross nodes are compromised, a th certificate detection is failure. The probability is $(c/N)^{a-1}$. So, we have

$$p = 1 - \left(\frac{c}{N}\right)^{a-1}. \quad (7)$$

The communication cost evaluation should consider two cases: a th detection is successful or failure. For the first case, communication cost includes query cost C_1 and copy cost C_2 . C_1 consists of the cost of transmitting proxy query request and the cost of sending two query requests from proxy query node. C_2 consists of the cost of transmitting proxy copy request and the cost of sending two copy requests from proxy query node. The hops at horizontal and vertical direction are, respectively, L and W . The average hops number between arbitrary two nodes is h . Consequently, we have

$$C = C_1 + C_2 = (h + W) + (h + L) = 2h + W + L. \quad (8)$$

If a th detection is failure, C is a sum of C_1 and the sent passive messages. If $a - 1$ cross nodes are uncompromised, $a - 1$ passive messages will respond to node N_i . In this case, $C = h + W + (a - 1)h$. So, we have

$$C = (1 - p)(2h + L) + W + p(ah + W). \quad (9)$$

In addition, the storage cost of CL method is

$$M = (a - 1)L + (1 - p)L = (a - p)L. \quad (10)$$

3.3. Security and Performance Evaluation. Data access control is necessary to ensure authorized access, since illegal access to sensitive data may cause disastrous consequences. From the view of protected object, security evaluation of data access control is classified into accessor security and access object security. The implementation of our method is analyzed as follows.

- (1) Effective access control: the master key in each stage is realized by encryption of a set of characteristics. Since the key chain is one-way, attackers cannot obtain the key for data encryption without the master key. Encryption on the master key has provable security under the BDH hypothesis. It demonstrates that attackers cannot decrypt the master key except that they have expectant access structure. Therefore, the method makes the data only be accessed by authorized user.
- (2) Constraining the collusion attack: the collusive users want to obtain the master key for data decryption. Actually, the method has provable security to select message attack under BDH hypothesis. The master key is encrypted as $Ke(g, g)^{ys}$. User can get K only by eliminating $e(g, g)^{ys}$. The only way to construct $e(g, g)^{ys}$ is $e(g^{(y-b)/\beta}, g^{\beta s}) = e(g, g)^{ys} / e(g, g)^{bs}$. Then, $e(g, g)^{bs}$ can be calculated. For each user, b is randomly selected from Z_p . The key of an unauthorized user is no use for other users to calculate $e(g, g)^{bs}$.
- (3) Limit impact of node capture: each sensor node only stores the key for current data encryption. The key used before will be erased. Due to the one-way feature of key, attackers cannot deduce previous keys by using current key. Each node encrypts data independently, which is not useful to capture other nodes.
- (4) Performance and functionality analysis: a sensor node is responsible for the following operations: (1) generate and encrypt the master key with the proposed method; (2) generate the key for data encryption by using the master key; (3) encrypt data of sensor nodes. These operations are deployed to various stages. Concretely, one node at one stage performs at most one dot product of scalar at elliptic curve, a one-way hash algorithm, and a symmetry data encryption.

In data request procedure, each node responds to data $\langle C_v, \{D\}_{K_i} \rangle$ at t th period of v th stage. C_v includes $f_i + 1$ group members in G_1 and a group element in G_T . $\{D\}_{K_i}$ is data

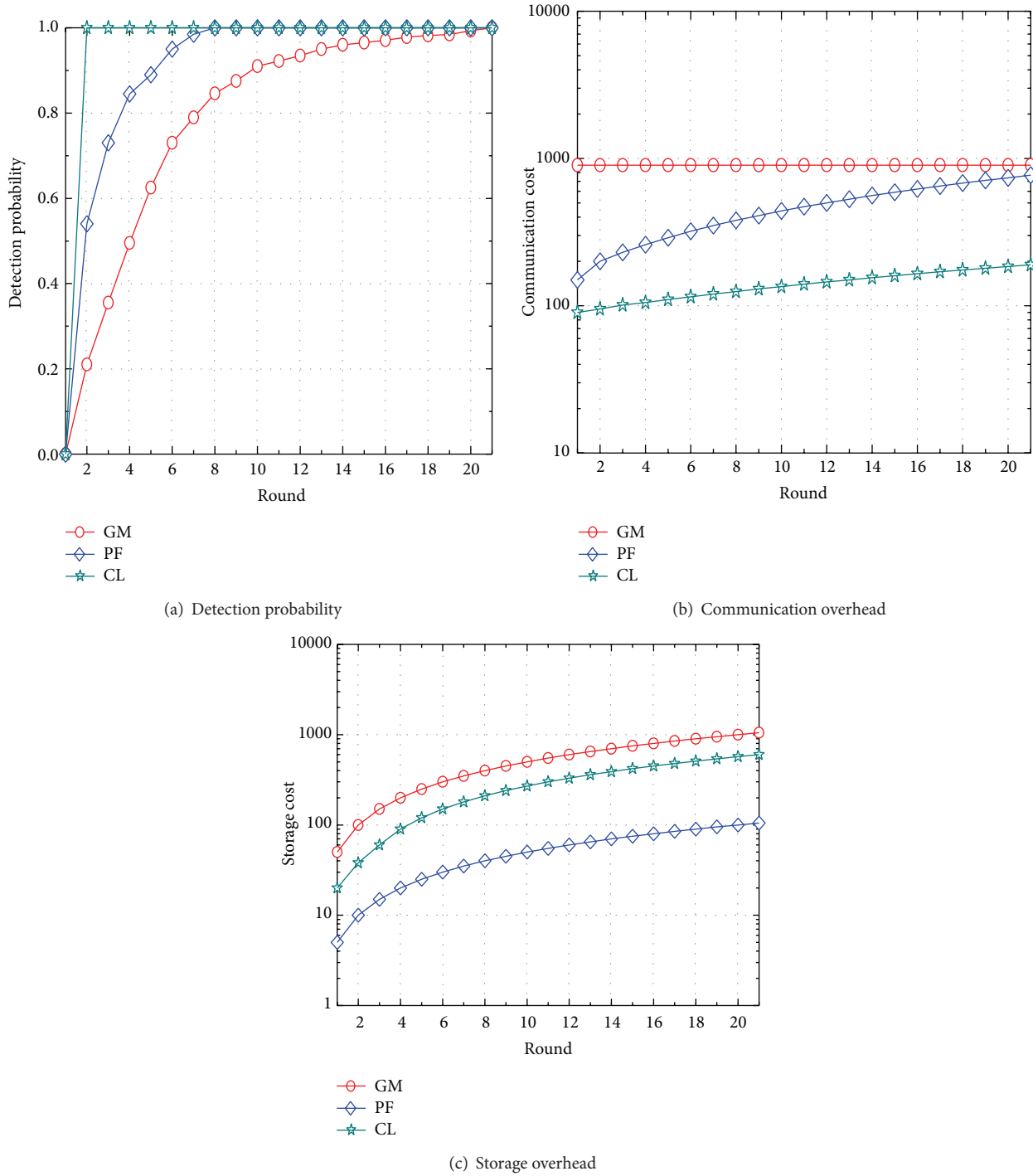


FIGURE 2: The relationship of detection rounds with detection probability, communication overhead, and storage overhead.

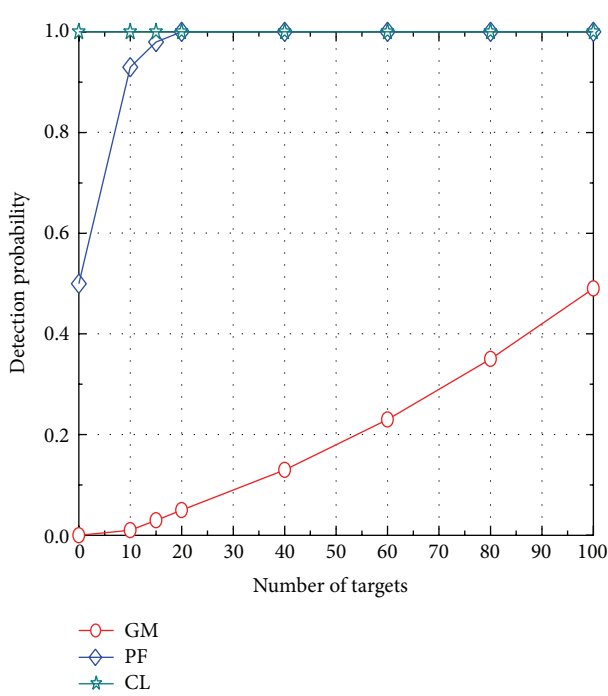
segment. In data cancel process, TP only needs to broadcast a group element in G_2 to all of sensor nodes.

Table 1 compares the functionalities of our method to those of BSW [8] and CN [10]. BSW has only designed threshold by simple combination of keys without network scalability and user revocation, and it can not withstand collusion attacks. CN is resilient to collusion attack and can resist some other attacks. The proposed method has

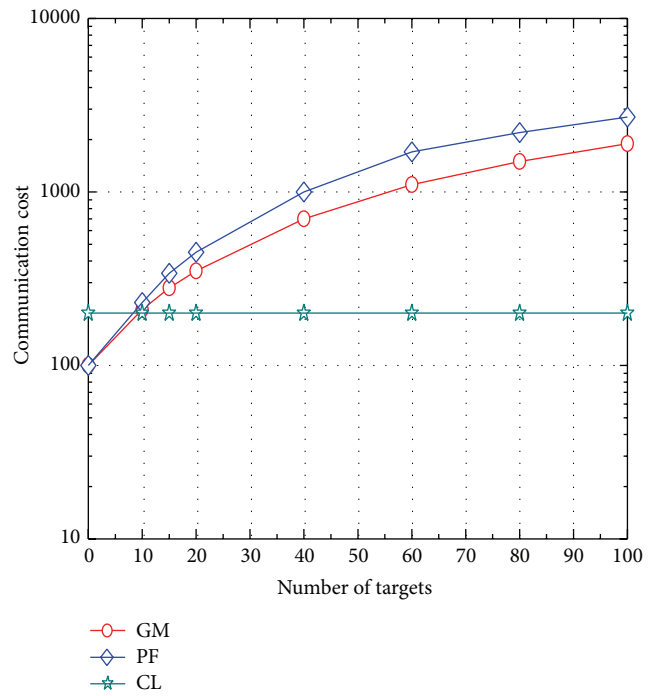
scalability, ability against collusion attack, and user cancel. Meanwhile, the descriptiveness is better and functionality is more comprehensive.

4. Experiments and Analysis

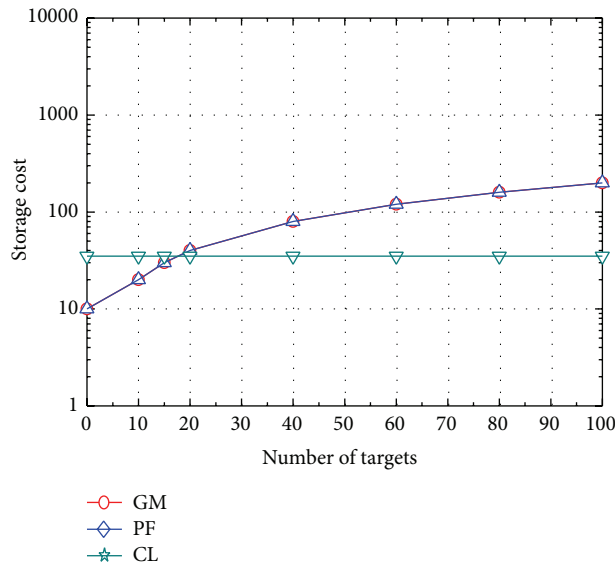
Simulations have been performed in NS-2 (Network Simulator version 2), developed by UC Berkeley University, to



(a) Detection probability



(b) Communication overhead



(c) Storage overhead

FIGURE 3: The relationship of witness nodes with detection probability, communication overhead, and storage overhead.

TABLE 1: Functionality comparison of various methods.

Method	Scalability	Descriptiveness	Ability against collusion attacks	User cancel
Proposed	No	Normal	No	No
CN	No	Bad	Yes	No
CEB	Yes	Good	Yes	Yes

evaluate the efficiency of the proposed scheme. The settings for experiments are as follows. There are 1000 nodes in WSN deployed in the area with the size of 1000×1000 . The communication radius of nodes is 50 m. The numbers of witness nodes (b) in GM and PF are, respectively, 50 and 5. The reason of choosing different b lies in their different detection probability. The number of compromised nodes is set to 100. Assume that no package loss or conflict occurs in data transmission. In Figure 2, every point represents the average usage of a certificate among 100 random nodes. For the use of each certificate, we randomly capture nodes for 100 times and calculate the average value. In addition, different network topology and random certificate are simulated. The evaluation indicators include detection probability, communication cost, and storage cost [15].

Figure 2(a) compares the detection probability of three methods as a function of the round of certificate used. As seen, the detection probability increases as a . When a is greater than 2, it can be completely detected. Since CL selects two cross curves to store and verify certificate, if the cross point is not captured, detection will be successful. But the detection probability is the minimum. On the contrary, the selection of target nodes in GM is totally random. It is not the optimized way for selection since a part of nodes may be captured, and certificate usage can be detected more than 8 times. By comparing to GM scheme, PF can realize feedback by using nodes on the path. Since b nodes will produce b paths, it is unable to capture nodes on all paths. Consequently, the detection accuracy is higher than that of GM scheme while requiring less witness nodes.

Figures 3(a), 3(b), and 3(c) compare the communication overhead and storage overhead of three detection schemes, separately. As we can see, the communication overhead in CL is lower than that of PF, but the storage overhead is opposite. In GM scheme, communication and storage overhead are larger than those of PF. This is because the witness nodes in GM are fixed and the number of witness nodes is more than that in PF. Since PF depends on other nodes and witness nodes on the path and requires less witness nodes, the communication and storage overhead are lower than those of GM.

Figure 3 demonstrates the effect of b on GM and PF. Due to the fact that we concern certificate use at the first time, $a = 2$ is set. Additionally, b has no impact on CL scheme. For comparison, it is also shown. As seen, the detection probabilities, respectively, in GM and PF are growing as b . The communication and storage overhead show the same trend. When $b = 10$, PF can detect the first abuse of certificate with the probability of 0.9. But in GM, the detection probability is less than 0.5, even $b = 100$. Since PF lets other nodes on the path responding to a passive message to the source node when detecting a message, it introduces higher communication overhead than GM. Moreover, when b is larger than 20, the detection probability of PF is close to 1. The communication and storage overhead are grown continuously as well. Consequently, a compromise should be chosen among detection probability, communication overhead, and storage overhead.

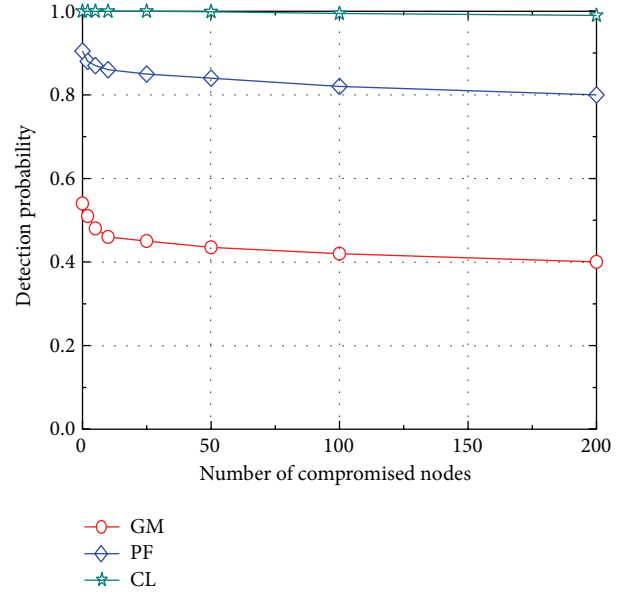


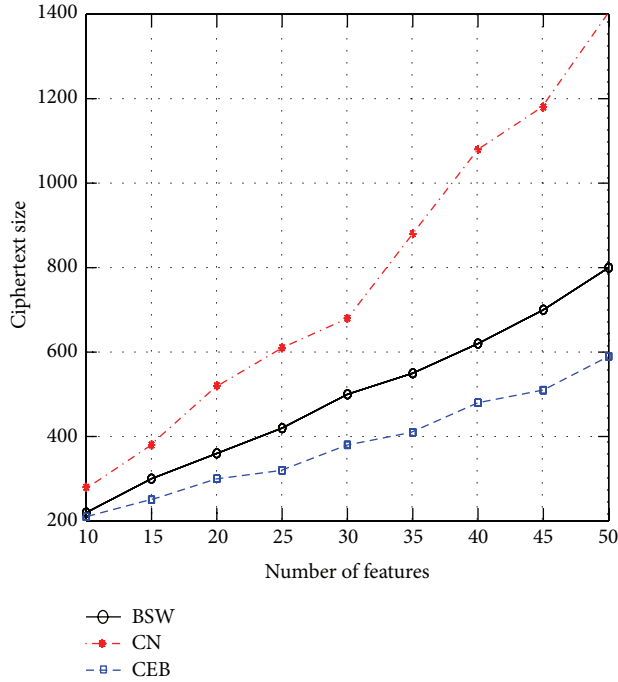
FIGURE 4: The relationship between detection probability and compromised nodes.

Figure 4 shows the detection probabilities in three schemes with increase of compromised nodes. We set $a = 2$ and $b = 50$ in GM and $b = 10$ in PF. In Figure 3, three schemes are affected by c since the number of witness nodes is random. For instance, CL can detect certificate being abused firstly with the probability of $p = 98\%$. Because many nodes on copy path may receive detection request message, it demonstrates that the scheme has good effect on capturing attacks.

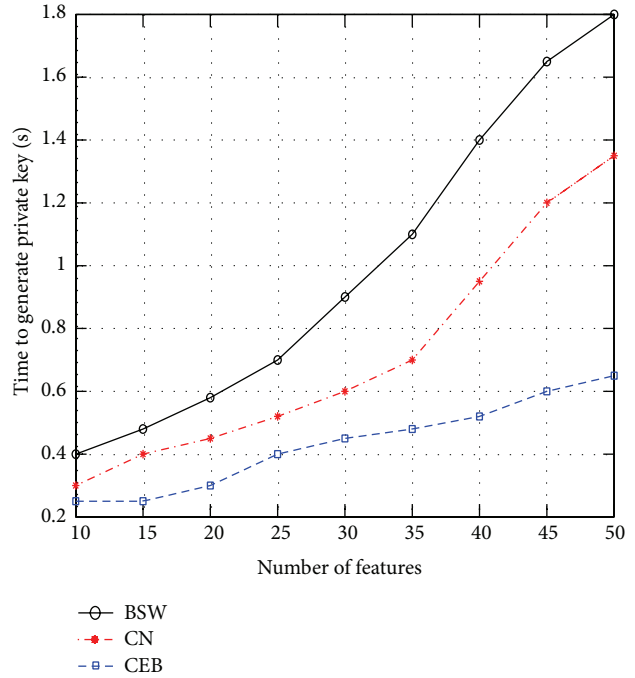
In the following, we simulate various performance indicators after applying our scheme, including length of ciphertext, key generation time, and time overhead on encryption and decryption. These indicators are compared to BSW and CN. In finite field, a super singular ellipse curve $y^2 = x^3 + x$. The time to match with PCB library is 5.5 ms. The time of selection random elements from G_1 and G_2 are 16 ms and 1.6 ms, respectively.

Figures 5(a)–5(d) show the comparison with length of ciphertext, key generation time, encryption time, and decryption time. Ciphertext includes ID, head, and data block. The head consists of characteristic collection f , a group element in G_2 , and $|f|$ group elements in G_1 .

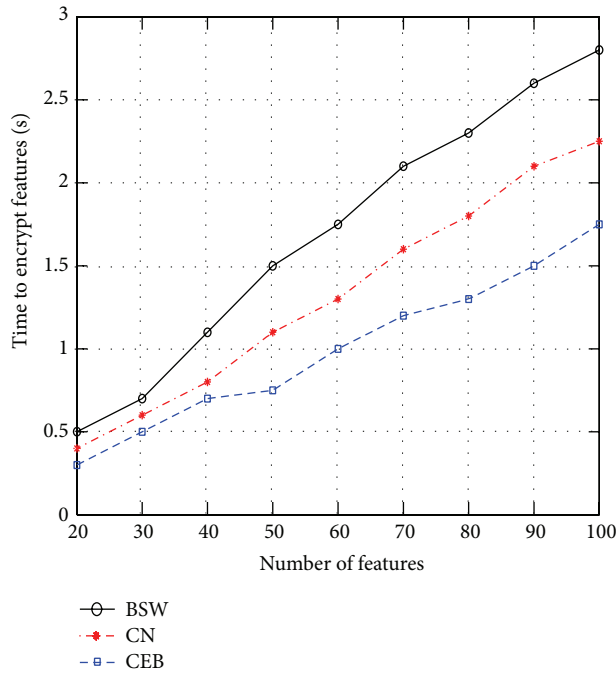
As we can see, the length of ciphertext, key generation time, and encryption time linearly increase as the number of characteristics in all three schemes. But the decryption time is not linear. Since the decryption time is related to the number of characteristics and access trees, different access tree may include different access structure. Moreover, the indicators in CEB are superior to that of other schemes. Because secret sharing is employed in encryption of BSW, serious access control is realized. Polynomial interpolation is required to construct key. Many complex matching and exponentiation operations are required in decryption. Although CN scheme replaces secret sharing with random elements in encryption, the length of ciphertext and key linearly increase as the



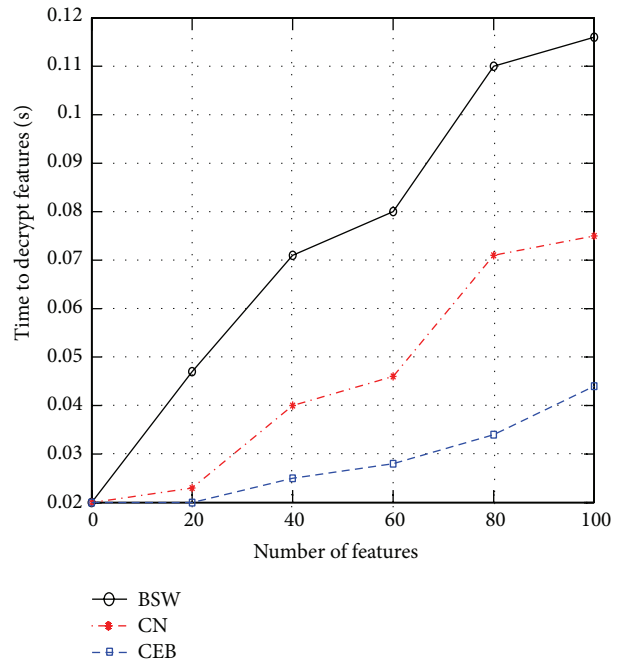
(a) Length of ciphertext



(b) Key generation time



(c) Encryption time



(d) Decryption time

FIGURE 5: The comparison of various schemes.

number of system characteristics. It causes low efficiency of CN scheme. However, our scheme encrypts and stores data periodically. The data of each node is encrypted by symmetric encryption algorithm. Meanwhile, the keys in encryption are linked as a one-way key chain. One key is used in a period.

The abilities against collusion attacks of three schemes are compared in Figure 6. Simply, we assume each node only

generate one data unit at each phase of per round. The total number of users in current network is supposed to be 100. The collusion users vary from 10 to 50. Then we simulate data leaking rate of three schemes.

As a note, the purposes of collusion users are to obtain more data with the key. By comparing with directly capturing node and intercepting attacks, collusion can save overhead

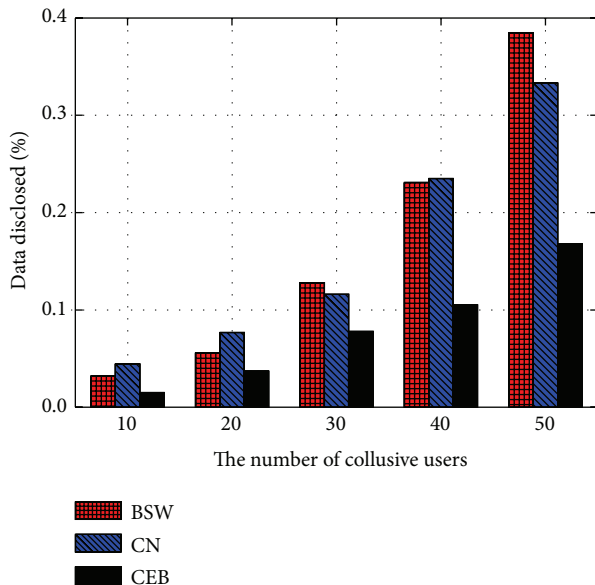


FIGURE 6: The comparison of abilities against collusion attacks for three schemes.

and is hard to be detected [16]. As shown in Figure 6, three schemes have almost the same abilities against attacks when there are few collusion users. With the growth of collusion users, data security of BSW and CN decreases rapidly, while that of our scheme shows a gentle decline. When the number of collusion users is greater than 40, data security of BSW scheme drops more dramatically than that of CN. As mentioned above, although CN generates key by utilizing the way of random number, actually, the number belongs to pseudorandom number. Attack can be realized with exhaustive method, easily with more collusion users. The proposed scheme has eliminated the above advantages. The master key is continuously updated. Once malicious users are found, we will cancel the operation. In this case, collusion users can only obtain their own data without obtaining data of other nodes. Therefore, the interference caused by attacks will be restricted to be the minimum.

5. Conclusion

To address the issue on security of access object and accessor, we proposed a digital certificate based remote data access control scheme. It is founded on access control scheme with characteristic expression. Our scheme has two features. On one hand, the network data is divided by characteristic and connected with key. When user requires query, the access control strategy, which is related to key, will judge the validity of the query. In this way, data access control is realized. On the other hand, anonymous authentication is realized for security of accessor. Moreover, three distributed certificate detection methods are designed for preventing certificate being abused by malicious anonymous users. The security analysis and experiments show that our scheme has the ability against collusion attacks and higher detection probability. The next

work is to perfect our scheme and apply it on real sensor nodes. The experiments are summarized as follows. GM has lower detection probability, or higher communication and storage overhead are required to achieve some detection probability. PF has higher detection probability but grows as witness nodes and successful use times of certificate. Meanwhile, the communication and storage overhead of PF are within a reasonable range. The case of CL scheme is similar to that of PF.

Conflict of Interests

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

Acknowledgments

This work is supported by the China Postdoctoral Science Foundation funded project (Grant no. 140778), the Natural Science Foundation of Fujian Province (Grant no. 2014J05079), the Young and Middle-Aged Teachers Education Scientific Research Project of Fujian province (Grant nos. JA13248 and JA14254), the special scientific research funding for colleges and universities from Fujian Provincial Education Department (Grant no. JK2013043), the Scientific Research Project of Minjiang University (Grant no. YKQ13003), and the Research Project supported by Xiamen University of Technology (YKJ13024R, XYK201437).

References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [2] P. Levis, S. Madden, J. Polastre et al., "Tinyos: an operating system for sensor networks," in *Proceedings of the 6th International Conference on Mobile Data Management (MDM '05)*, pp. 115–148, IEEE, Nara, Japan, 2005.
- [3] S. Bhatti, J. Carlson, H. Dai et al., "Mantis os: an embedded multithreaded operating system for wireless micro sensor platforms," *Mobile Networks and Applications*, vol. 10, no. 4, pp. 563–579, 2005.
- [4] C. Han, R. Kumar, R. Shea et al., "A dynamic operating system for sensor networks," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys '05)*, pp. 163–176, Seattle, Wash, USA, 2005.
- [5] C. Karlof and D. Wangner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 113–127, IEEE, Anchorage, Alaska, USA, May 2003.
- [6] B. Carbutar, Y. Yu, L. Shi, M. Pearce, and V. Vasudevan, "Query privacy in wireless sensor networks," in *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '07)*, pp. 203–212, June 2007.
- [7] D. He, J. Bu, S. Zhu, S. Chan, and C. Chen, "Distributed access control with privacy support in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3472–3481, 2011.

- [8] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the IEEE Symposium on Security and Privacy (SP '07)*, pp. 321–334, Berkeley, Calif, USA, May 2007.
- [9] H. Wang and Q. Li, "Distributed user access control in sensor networks," in *Proceedings of the International Conference on Distributed Computing in Sensor Systems (DCOSS '06)*, pp. 305–320, Berlin, German, 2006.
- [10] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 456–465, Alexandria, Va, USA, November 2007.
- [11] Z. Ruan, W. Liang, D. Sun, H. Luo, and F. Cheng, "An efficient and lightweight source privacy protecting scheme for sensor networks using group knowledge," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 601462, 14 pages, 2013.
- [12] Z. Ruan, X. Sun, and W. Liang, "Securing sensor data storage and query based on k -out-of- n coding," *International Journal of Communication Systems*, vol. 26, no. 5, pp. 549–566, 2013.
- [13] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*, pp. 243–254, ACM, Boston, Mass, USA, August 2000.
- [14] Z.-W. Tan, Z.-J. Liu, and C.-M. Tang, "A proxy blind signature scheme based on DLP," *Journal of Software*, vol. 14, no. 11, pp. 1931–1935, 2003.
- [15] R. Sarkar, X. Zhu, and J. Gao, "Double rulings for information brokerage in sensor networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 6, pp. 1902–1915, 2009.
- [16] N. Subramanian, K. Yang, W. Zhang et al., "Ellips: privacy preserving scheme for sensor data storage and query," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '09)*, pp. 936–944, Rio de Janeiro, Brazil, 2009.

Research Article

Parking Query in Vehicular Delay-Tolerant Networks with Privacy Protection Based on Secure Multiparty Computation

Haiping Huang,^{1,2,3,4} Juan Feng,^{1,2} Dan Sha,^{1,2} Jia Xu,^{1,2,3} and Hua Dai^{1,2,4}

¹College of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

²Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing 210003, China

³Key Lab of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

⁴College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Correspondence should be addressed to Haiping Huang; hhp@njupt.edu.cn

Received 18 October 2014; Revised 19 February 2015; Accepted 11 March 2015

Academic Editor: Fei Yu

Copyright © 2015 Haiping Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Within vehicular delay-tolerant network, conflict exists in the scenario of which two vehicles happen to choose the same parking space. To solve this problem, two protocols are proposed, respectively, which are called privacy protection protocol based on secure multiparty computation and routing protocol based on angle and density. The proposed methods prevent the leaking of privacy information of the vehicles involved during the parking space seeking process and improve the performance of the transmission ratio and reduce the transmission delay by unifying the directions of messages and choosing the vehicle of the highest distribution density as the next hop. The results of the simulation show the efficiency of our method.

1. Introduction

Many applications under the delay-tolerant network have been studied and extended to wireless network domains such as mobile ad hoc networks (MANET), vehicular ad hoc networks (VANET), and wireless sensor networks (WSN). In the VANET, communications between two vehicles are intermittent due to the changes of the driving speed and direction.

One key issue in VANET is the urban parking problem. It arises along with the rapid expansion of urban population. The intelligent parking space positioning system is developed to address this issue. In this paper, we assume a situation where two nearby vehicles happen to find the same parking space which causes a competition for the limited resource.

To solve the resource allocation problem, we propose a multiparty computation based protocol to detect the potential conflict. Once the vehicle detects that the parking space has been already allocated, it continues its searching process. Vehicular delay-tolerant network is full of potential

maliciousness, such as attacks and unauthorized operations. The vehicle privacy is also a critical issue. In order to protect it, we design a privacy protection strategy during the secure multiparty computation, along with a routing protocol to increase the transmission ratio while decreasing the transmission delay. The distribution density and the directions of the vehicles are input parameters in estimation function. The direction of the vehicle is calculated by measuring the angle between the reference direction and the line linking itself to the base station. The vehicle near this line has the highest distribution density and will be chosen for the next hop.

In this paper, we brief the related work on security and routing protocol of delay-tolerant networks in Section 2. The privacy protection strategy based on multiparty computation is designed in Section 3, while the routing protocol based on angle and density is proposed in Section 4. In Section 5, we discuss the security of the proposed method, and the simulation results show the higher transmission ratio and lower transmission delay. Section 6 gives the conclusion of this paper.

2. Related Work

The communication interruption and the uncertain network topology are the main reasons that leave the security of delay-tolerant network a challenging issue. However, some existing application system fails to meet the basic requirements for security purpose. Ensuring the security in the delay-tolerant network is an active topic among researchers. Lu et al. [1] proposed a privacy protection strategy by using the filter. The filter is maintained by the corresponding node according to its own interest. Hur and Kang [2] proposed a secure data retrieval scheme that used the CP-ABE for decentralized delay-tolerant networks, and multiple key authorities manage their own attributes. Lv et al. [3] designed an efficient and noninteractive key exchange protocol based on a time-evolving topology model and a two-channel cryptography. A time-evolving model is used to formally analyze the periodic and predetermined behavior patterns, and therefore a node can schedule when and to whom it should send its public key. By analyzing the periodic and predetermined behavior patterns, the time-evolving model decided and prepared the receiver of the node for its public key. The third party methods have drawn the attention of many researchers in delay-tolerant network. Rane et al. [4] presented a scheme using biometric authentication and homomorphic encryption for secure calculation of Hamming distance while protecting the participants' privacy. And they furthered their study by proposing a secure two-party computation protocol [5] of Euclidean distance using Paillier homomorphic encryption and this protocol is implemented for private querying of face images and maintains low communication overhead. Facial information is utilized for authentication, and communication overhead is kept as a minimum. Yang et al. [6] provided a quantum privacy comparison method. The presence of a semihonest third party allowed the matching of information without revealing the nodes' privacy. Huang et al. [7] allow the participants to calculate the distance between each other based on an honest third party in order to realize comparison of local data. However, the private values of participants are fully grasped by the third party, which is followed by obvious information leakage and security issues that cannot be ignored. Gao et al. [8] encoded the original data for privacy purpose and select the top N nodes to perform the multicast in network by applying privacy protected data forwarding (PPDF) model.

The routing protocol for delay-tolerant network is an active topic in many different contexts. Samuel et al. [9] introduced the dominant set to the routing protocol. They took Markov model to predict the distribution of encountering interval and evaluate the utility value for the next hop. According to the historical statistic data of vehicle routes, Xu et al. [10] suggested the use of statistic method for encountering probability prediction. However, the statistic data might fail in the real scenario. Dunbar and Qu [11] applied the statistic method to preventing the location information of vehicles leaking by communication with RSUs. Hui et al. [12] binned the nodes into several areas according to their encountering probabilities simultaneously taking into account the community attribute and center degree. Fabbri and Verdone [13] considered the characteristic curve

of social relations against the time series to plan the routing. Lee et al. [14] extended the current geographical routing protocol by forming a two-level hierarchy for heterogeneous network. However, existing methods either are computationally expensive or are having considerable network delay, not to mention the security issue.

In this paper, we design a secure multiparty computation protocol for privacy protection. By using the direction and distribution density to decide the next hop, higher transmission ratio and lower transmission delay have been achieved.

3. Privacy Protection Protocol Based on Multiparty Computation

In order to find conflict of the same parking spaces, vehicles must take part in multiparty computation and meanwhile have to exchange the already owned parking space information with others. As a result, a vehicle may excavate some privacy information of other vehicles, and obviously it is dangerous. In the proposed protocol, as the third party, the base station instead of the vehicles will finally calculate the common parking spaces. Space information of each vehicle will be packed with polynomial function before being sent, and then the base station and other vehicles taking part in calculation cannot acquire the real space information. So privacy is protected. In this paper, we consider a scenario where there are three vehicles searching for parking spaces within a certain area.

Firstly, each vehicle participating in calculation will search for the parking spaces within a certain communicated area and then generate a set according to the parking spaces discovered by it. Secondly, it will generate the special polynomial function. Thirdly, it will send the function to the base station. Finally, the base station will figure out the common parking spaces and return the result.

3.1. Problem Description. Assuming that there are three vehicles within a certain communication area, for example, A, B, and C, the sets they generate according to the parking spaces they have found are $S_a = \{a_1, a_2, a_3, \dots, a_l\}$, $S_b = \{b_1, b_2, b_3, \dots, b_m\}$, and $S_c = \{c_1, c_2, c_3, \dots, c_n\}$, respectively. The three vehicles expect to deduce common parking spaces without leaking out their private information during the computing process.

3.2. Description of the Protocol. Vehicle A and vehicle B generate polynomial functions according to $S_a = \{a_1, a_2, a_3, \dots, a_l\}$ and $S_b = \{b_1, b_2, b_3, \dots, b_m\}$, denoted by $f(x)$ and $g(x)$ separately:

$$\begin{aligned} f(x) &= (x - a_1) \times (x - a_2) \times (x - a_3) \times \dots \times (x - a_l), \\ g(x) &= (x - b_1) \times (x - b_2) \times (x - b_3) \times \dots \times (x - b_m). \end{aligned} \quad (1)$$

And then vehicle C will calculate $f(c_i)$ and $g(c_i)$; if $f(c_i) = 0$ and $g(c_i) = 0$, then element c_i must be in sets S_a and S_b ; in other words,

$$f(c_i) + g(c_i) = 0 \wedge f(c_i) \times g(c_i) = 0. \quad (2)$$

It is equivalent to the fact that

$$f(c_i) = 0 \wedge g(c_i) = 0. \quad (3)$$

So, $c_i \in S_a \wedge c_i \in S_b$, which means that element c_i is one of the common parking spaces of vehicles A, B, and C.

3.3. Implementation of the Protocol. (1) Vehicles A, B, and C send request message to the base station, respectively, telling the base station that they want to inquire common parking spaces.

(2) After the base station receives the request messages, it sends response message to vehicles A, B, and C, and then connection is established.

(3) Vehicle A generates polynomial function as follows:

$$\begin{aligned} f(x) &= (x - a_1) \times (x - a_2) \times (x - a_3) \times \cdots \times (x - a_l) \\ &= \sum_{i=0}^l p_i x^i = \sum_{i=0}^l q_i x^i + \sum_{i=0}^l r_i x^i, \end{aligned} \quad (4)$$

where q_i, r_i ($i = 1, 2, \dots, l$) are all nonzero real numbers.

In addition, we assume that $f_b(x) = \sum_{i=0}^l q_i x^i$, $f_c(x) = \sum_{i=0}^l r_i x^i$, $f_b(x) : A \rightarrow B$, and $f_c(x) : A \rightarrow C$.

(4) Vehicle A then sends $f_b(x)$ and $f_c(x)$ to the base station using the routing protocol we proposed in the following sections.

(5) Vehicle B generates polynomial function as follows:

$$\begin{aligned} g(x) &= (x - b_1) \times (x - b_2) \times (x - b_3) \times \cdots \times (x - b_m) \\ &= \sum_{i=0}^m e_i x^i = \sum_{i=0}^m s_i x^i + \sum_{i=0}^m t_i x^i, \end{aligned} \quad (5)$$

where s_i, t_i ($i = 1, 2, \dots, l$) are all nonzero real numbers.

In addition, we assume that $g_a(x) = \sum_{i=0}^m s_i x^i$, $g_c(x) = \sum_{i=0}^m t_i x^i$, $g_a(x) : B \rightarrow A$, and $g_c(x) : B \rightarrow C$.

(6) Vehicle B then sends $g_a(x)$ and $g_c(x)$ to the base station using the routing protocol we proposed in the following sections.

(7) The base station finally receives value of $f_b(x)$, $f_c(x)$, $g_a(x)$, and $g_c(x)$; then it calculates $h_c(x)$ and the parameter 0.5 is considered according to previous simulation results or experience:

$$h_c(x) = f_b(x) \times g_c(x) + 0.5 \times g_a(x) \times f_b(x). \quad (6)$$

(8) The base station then sends $g_a(x)$ to vehicle A.

(9) After receiving $g_a(x)$, vehicle A calculates $u(x)$ and $v(x)$:

$$\begin{aligned} u(x) &= f(x) + g_a(x) = \sum_{i=0}^j \alpha_i x^i \quad (j = \max(l, m)), \\ v(x) &= g_a(x) \times f_c(x) + 0.5 \times g_a(x) \times f_b(x) \\ &= \sum_{i=0}^k \beta_i x^i \quad (k = l + m). \end{aligned} \quad (7)$$

(10) Vehicle A then generates private vectors on the basis of (7) as follows:

$$\begin{aligned} \mathbf{X}_1 &= (\alpha_0, \alpha_1, \dots, \alpha_j), \\ \mathbf{X}_2 &= (\beta_0, \beta_1, \dots, \beta_k). \end{aligned} \quad (8)$$

Afterwards, vehicle A sends vectors \mathbf{X}_1 and \mathbf{X}_2 to the base station.

(11) The base station packs \mathbf{X}_1 , \mathbf{X}_2 , $f_c(x)$, $g_c(x)$, and $h_c(x)$ and then sends the package to vehicle C.

(12) After receiving package of \mathbf{X}_1 , \mathbf{X}_2 , $f_c(x)$, $g_c(x)$, and $h_c(x)$, vehicle C then calculates $h(x)$:

$$h(x) = f_c(x) \times g_c(x) + h_c(x). \quad (9)$$

(13) Following that, vehicle C implements the following substeps using elements c_i ($i = 1, 2, \dots, n$) of set S_c .

(a) Generating private vectors:

$$\begin{aligned} \mathbf{Y}_1 &= (1, c_i, c_i^2, \dots, c_i^j), \\ \mathbf{Y}_2 &= (1, c_i, c_i^2, \dots, c_i^k). \end{aligned} \quad (10)$$

(b) Computing vectors \mathbf{U}_1 and \mathbf{U}_2 :

$$\begin{aligned} \mathbf{U}_1 &= \mathbf{X}_1 \times \mathbf{Y}_1, \\ \mathbf{U}_2 &= \mathbf{X}_2 \times \mathbf{Y}_2. \end{aligned} \quad (11)$$

(c) Calculating $g_c(c_i)$ and $h(c_i)$, judging whether $\mathbf{U}_1 + g_c(c_i) = 0$ and $\mathbf{U}_2 + h(c_i) = 0$ in the meantime. If it is true, then $c_i \in (S_a \cap S_b)$; otherwise, $c_i \notin (S_a \cap S_b)$.

(14) Vehicle C will finally obtain all elements c_i which satisfy $c_i \in (S_a \cap S_b)$, and then it sends $S_u = \{c_i, c_i \in (S_a \cap S_b), i = 1, 2, \dots, n\}$ to the base station.

(15) The base station sends the common parking spaces set S_u to vehicle A and vehicle B. Eventually, the three vehicles acquire their common parking spaces without obtaining private information of others.

4. Routing Protocol Based on Angle and Density

During the calculation of common parking spaces, parking information is transmitted in the network, so we propose a routing protocol for vehicles to decide how to choose routes along to the base station when sending messages.

In vehicular delay-tolerant networks, wireless communication devices are equipped with vehicles. When the distance of two vehicles is shorter than the communication radius, they can connect and exchange messages. While in some blind areas or in some periods of time, there may be no continuous connections between the source vehicle and the destination vehicle due to the sparse distribution, signal shielding, and high speed mobility of vehicles, which are the fundamental characteristics of delay-tolerant networks, also

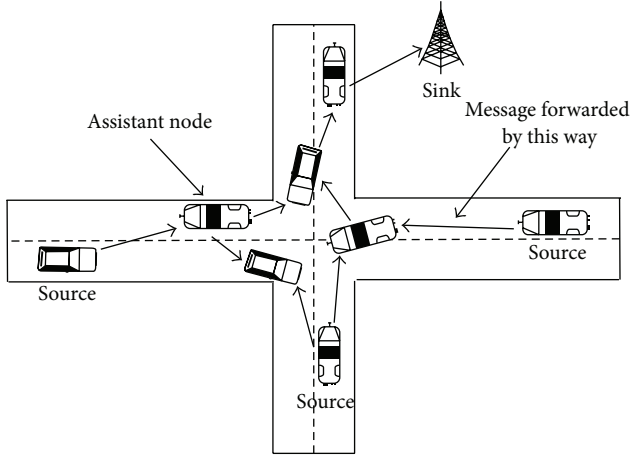


FIGURE 1: Network model of vehicular delay-tolerant networks.

called intermittent connectivity. If a vehicle wants to send a message, it will firstly carry this message, until it encounters the other vehicle, and then it forwards the message. Due to the “storage-carry-forward” scheme, transmission delay is always an important issue. Consequently, research of routing algorithms and message distribution algorithms is significant to reduce transmission delay.

As described above, considerable literatures have studied the characteristics of transmission delay. Liu et al. [15] analyze transmission delay in vehicular delay-tolerant networks with a bidirectional traffic model and it is demonstrated that transmission delay is linearly associated with transmission distance and subsequently indicates that a certain relation exists between transmission delay and vehicle density in the road. The research conclusion will be used in this paper.

4.1. Network Model. Figure 1 manifests the network model of vehicular delay-tolerant networks, and message is forwarded by assistant node from the source nodes to the base station. In other words, message is transmitted from one vehicle to another vehicle in the range of communication.

Different vehicles, as a general rule, may have different speeds; for example, the vehicle in the fast traffic lane usually runs faster than those in the slow traffic lane. We can set n levels of the vehicle speed, from slow to fast, which are denoted by v_1, v_2, \dots, v_n ($v_i, i = 1, 2, \dots, n$), respectively. For convenience sake, we firstly consider vehicles traveling in the same direction. We can assume the set $S_i = \{x_{i,j}(t), j \in N\}$, ($i = 1, 2, \dots, n$), is the set of vehicles traveling at the speed v_i , where $x_{i,j}(t)$ indicates the j th vehicle of the set S_i at time t and t is discrete, $t \in N$, and N is the natural number.

Assume the relationship of vehicle speed v and vehicle density λ is as follows:

$$v = a - b\lambda; \quad (12)$$

when $\lambda = 0$, the vehicle density is 0, and then the vehicle speed v can achieve the largest value v_f ; that is, when $\lambda = 0$, then $a = v_f$. When $\lambda = \lambda_f$, it means the vehicle density achieves the largest value λ_f , and then the vehicle speed will

be 0 and the road is congested; that is, when $\lambda = \lambda_f$, then $b = v_f/\lambda_f$. Thus it can be seen that the relationship of vehicle speed v and vehicle density λ is

$$v = a - b\lambda = v_f - \frac{v_f}{\lambda_f}\lambda = v_f \left(1 - \frac{\lambda}{\lambda_f}\right). \quad (13)$$

4.2. Definition of Transmission Delay. In vehicular delay-tolerant networks, transmission delay is mainly caused by message retransmission due to communication breakdown or signal interference. While compared with the time of a message transmitted from one vehicle to the other vehicle, message retransmission delay is short enough. So we can ignore it.

Assuming that a vehicle N_1 broadcasts a message at the time $t = 0$, we can take $I(t)$ as the set of vehicles having received the message from N_1 at time t and take $T(k) = \inf\{t \geq 0 : x_k \in I(t)\}$ as the time when vehicle N_k receives the message, and then the transmission delay from vehicle N_p to vehicle N_q can be defined as follows:

$$D(p, q) = T(q) - T(p) = \sum_{k=p}^{q-1} [T(k+1) - T(k)]. \quad (14)$$

4.3. Details of Routing Protocol. In the routing protocol based on angle and density (abbreviated to RPAD) proposed in this paper, the density and direction of vehicle are used for estimate. When a vehicle wants to send message to the base station, it will compute the angle between itself and the base station in order to choose relay vehicles in the suitable direction along to the base station, and meanwhile it will confirm the next hop after predicting vehicle density.

Assuming vehicle O is the source node, we create a coordinate and take O as the center, shown in Figure 2. Vehicles A and B are in the communication range of vehicle O, and vehicle C is in the communication range of vehicle B. To reduce the transmission distance of a message and then to reduce the transmission delay, we need to unify the direction of the next hops and the message could be transmitted in the unified direction along to the base station as a guarantee of delivery ratio. Vehicle O obtains the angle between itself and the base station by the GPS device it equips, and if the angle value is larger than 45° , then it will choose those vehicles whose angle between themselves and the base station is smaller than 45° as the next hops. In Figure 2, the angle between vehicle B and the base station is $\angle 1 < 45^\circ$, while that between vehicle A and the base station is $\angle 2 > 45^\circ$, and then vehicle O will choose vehicle B as the next hop and sends message to vehicle B. Similarly, through the comparison of angles, vehicle B will choose vehicle C as the next hop. Consequently, the message of vehicle O will be forwarded along the direction like a stepped appearance shown in Figure 2 finally to the base station.

However, there may be many vehicles satisfying the condition of angle direction. When a vehicle decides the next hop, it cannot send a copy of the message to all the condition-satisfied vehicles for the purpose of energy-efficiency. It is found by studies that, in the area of high vehicle density,

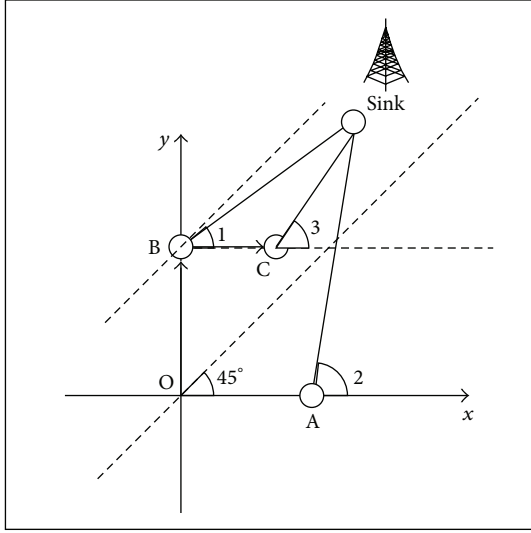


FIGURE 2: Coordinate of vehicle O.

a source vehicle can transmit messages rapidly step by step through other vehicles as a relay until the destination vehicle, instead of carrying the messages until the source vehicle encounters the destination vehicle. Thus, message can be delivered faster to the destination, and then transmission delay is reduced.

Yazhi demonstrates the following theorem [13] through theoretical analysis and simulation experiment.

Theorem 1. *There exists a relationship between the transmission delay $T_{0,m}$ and corresponding transmission distance m :*

$$\lim_{m \rightarrow \infty} \frac{T_{0,m}}{m} = \alpha. \quad (15)$$

Based on (15), when the vehicle density and speed are fixed, α is a constant:

$$\alpha = \lim_{m \rightarrow \infty} \frac{E(T_{0,m})}{m}. \quad (16)$$

Theorem 1 demonstrates that the limited ratio of transmission delay and transmission distance is α , and α is a constant when vehicle density is fixed, so transmission delay is linearly associated with transmission distance. The upper bound of the coefficient of the linear relation changes with vehicle density and speed, and so transmission delay decreases with the increase of vehicle density.

In RPAD, after selecting a direction, the vehicle will decide to transmit messages to those vehicles with higher density in their own areas.

Taking vehicle O for an example, $S_i = \{x_{i,j}(t), j \in N\}$, ($i = 1, 2, \dots, n$), is the set of vehicles traveling at the relative speed v_i . In the next time s , $S = \{x_{i,j}(t) + v_i(s - t), i = 1, 2, \dots, n, j \in N\}$ represents the relative distance between vehicles that satisfied the requirement of angle direction and vehicle O. Vehicle O will further choose those vehicles with higher area density as next hops according to the relative

distance and then sends messages to them. Other vehicles repeat the above process until the base station receives the messages.

5. Experiment Simulation Results and Analysis

5.1. Correctness Proof of Privacy Protection Protocol

5.1.1. *Certification.* It is obvious from Section 3 that

$$\begin{aligned} U_1 &= X_1 \times Y_1 \\ &= \alpha_0 + \alpha_1 \times c_i + \dots + \alpha_j \times c_i^j + g_c(c_i) \\ &= u(c_i) + g_c(c_i) = f(c_i) + g(c_i), \\ U_2 &= X_2 \times Y_2 \\ &= \beta_0 + \beta_1 \times c_i + \dots + \beta_k \times c_i^k + h(c_i) \\ &= v(c_i) + h(c_i) = f(c_i) \times g(c_i). \end{aligned} \quad (17)$$

If $U_1 = 0$ and $U_2 = 0$, then

$$\begin{aligned} f(c_i) + g(c_i) &= 0, \\ f(c_i) \times g(c_i) &= 0. \end{aligned} \quad (18)$$

Therefore, $f(c_i) = 0$ and $g(c_i) = 0$.

Based on (1), $c_i \in S_a$ and $c_i \in S_b$.

5.1.2. *Disproof.* If $c_i \in S_a$ and $c_i \in S_b$, then, with (1), $f(c_i) = 0$ and $g(c_i) = 0$.

While $U_1 = f(c_i) + g(c_i)$ and $U_2 = f(c_i) \times g(c_i)$, $U_1 = 0$ and $U_2 = 0$.

So $U_1 = 0 \cap U_2 = 0 \Leftrightarrow c_i \in S_a \cap S_b \cap S_c$.

In conclusion, the privacy protocol is proved to be correct.

5.2. *Security Proof of Privacy Protection Protocol.* Next, it will be proved that vehicles A, B, and C will finally obtain the information of common parking spaces without leaking out their respective private information of parking spaces.

(a) *Vehicles A and B.* Vehicle A has $g_a(x)$ and vehicle B has $f_b(x)$, while $g_a(x)$ is justly the part of $g(x)$ and $f_b(x)$ is the part of $f(x)$, and as a result vehicle A cannot get $g(x)$ from $g_a(x)$ and vehicle B cannot get $f(x)$ from $f_b(x)$. In other words, vehicle A will not obtain private parking space information of vehicle B, and so does vehicle B.

(b) *Vehicle C.* Vehicle C has $f_c(x)$ and $g_c(x)$, while $f_c(x)$ is justly the part of $f(x)$ and $g_c(x)$ is the part of $g(x)$. In addition, $h_c(x) = f_b(x) \times g_c(x) + 0.5 \times g_a(x) \times f_b(x)$. So vehicle C cannot get $f(x)$ and $g(x)$; that is, vehicle C will not obtain private parking space information of vehicles A and B.

Above all, private parking space information of the three vehicles has not been leaked out, so the protocol is secure.

5.3. *Performance Analysis of Privacy Protection Protocol.* Here, the complexity of communication round and computation is evaluated. Performance comparison results of some privacy protection schemes are shown in Table 1.

TABLE 1: Performance comparison of some privacy protection schemes.

	Communication round complexity	Computation complexity	Degree of privacy protection
Scheme in this paper	3	$O(n)$	Zero leakage
Shantanu's scheme [4]	n	$O(n^3)$	Zero leakage
Shantanu's promoted scheme [5]	n	$O(n^3)$	Zero leakage
Yang's scheme [6]	4	$O(n^2)$	Zero leakage
Huang's scheme [6]	3	$O(n^2)$	The TP obtains all messages from two participants

TABLE 2: The main simulation parameters.

Parameters	Value
Simulation time (hour)	10
Simulation range (width, height: meter)	80000, 58400
Number of vehicles	100–200
Vehicle speed (high, low: meter/s)	24, 6
Communication distance (meter)	150
Number of copies of SW/SF	10

Obviously from Table 1, the scheme proposed in this paper has the same communication rounds with Huang's scheme that is superior to other schemes whose communication cost is proportional to original vector dimensions n . Meanwhile, on the premise of privacy zero leakage, our scheme obtains the best computation complexity $O(n)$, owing to the lightweight mathematical method rather than traditional homomorphism encryption applied in other schemes.

5.4. Simulation Results and Analysis of Routing Protocol. We use ONE (opportunistic network environment) to realize the simulations on RPAD proposed in this paper. ONE is specially designed for simulations in delay-tolerant networks, having implemented some classical routing algorithms, like epidemic, SW, SF, and so on. Some main simulation parameters are described in Table 2.

In the experiment, the transmission ratio is defined as the ratio of a number of messages that have been transmitted successfully and that have been generated. Transmission delay is defined in Section 3.2. To evaluate the performance of these algorithms RPAD, SW, SF, and PER, we vary the vehicle speed and the number of vehicle nodes.

5.4.1. Influence of Vehicle Speed. Figures 3 and 4 indicate the influence of vehicle speed on transmission ratio and transmission delay, respectively. When the vehicle speed increases, the transmission ratio increases and transmission delay decreases, because when the vehicle runs at a higher speed, messages carried by vehicles are forwarded faster; that is, the time from the source node to the destination node will be shorter. It can be concluded from Figures 3 and 4 that RPAD performs better than SW, SF, and PER.

5.4.2. Influence of Total Number of Vehicle Nodes. Figures 5 and 6 indicate the influence of total number of vehicles on

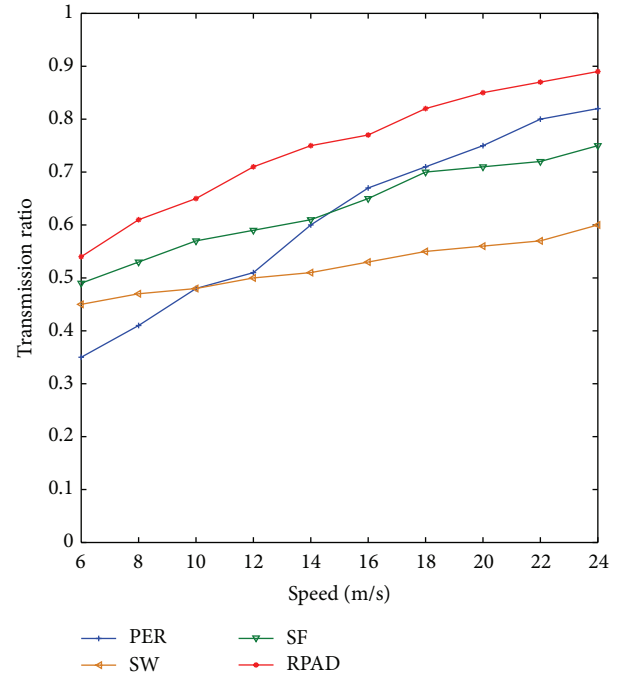


FIGURE 3: The influence of vehicle speed on transmission ratio.

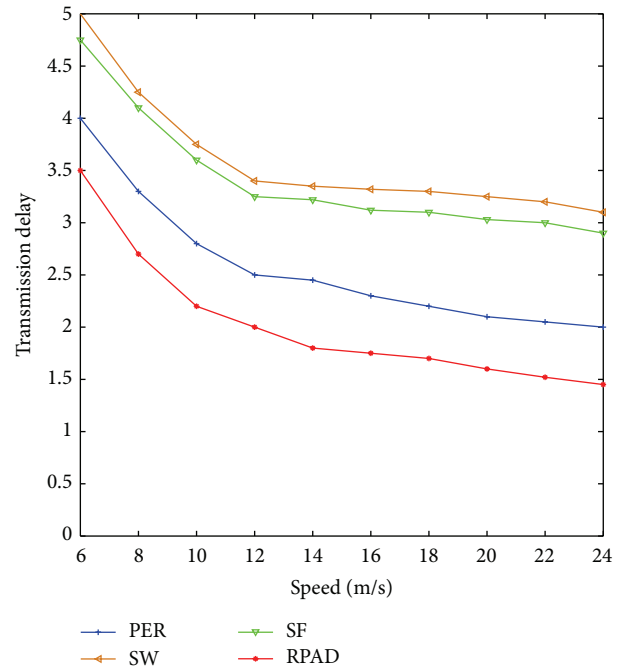


FIGURE 4: The influence of vehicle speed on transmission delay.

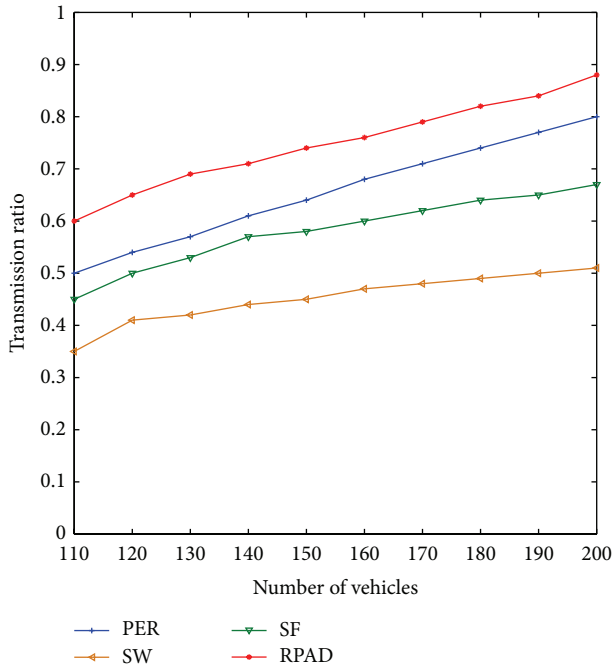


FIGURE 5: The influence of number of vehicles on transmission ratio.

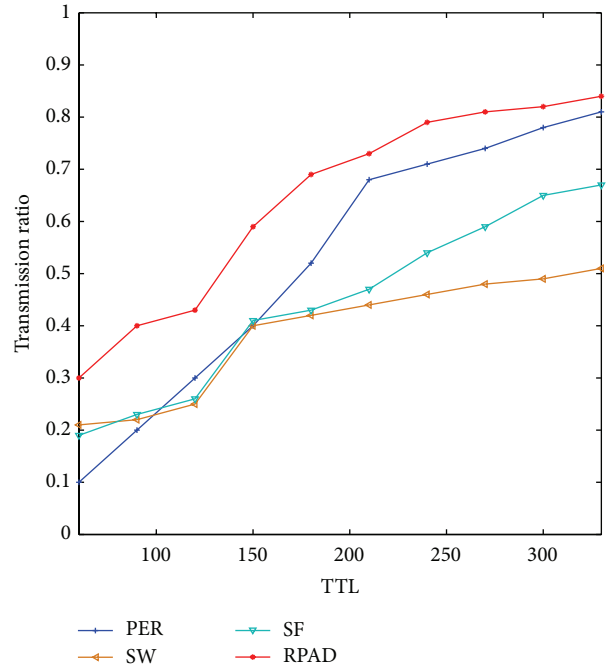


FIGURE 7: The influence of TTL on transmission ratio.

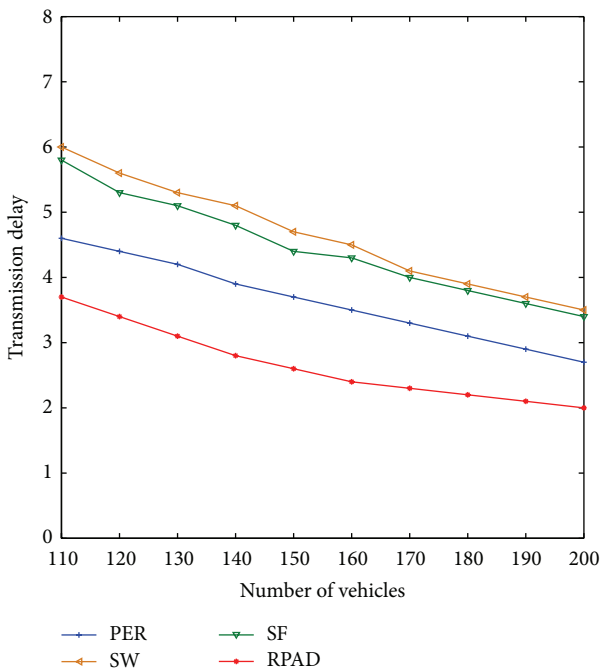


FIGURE 6: The influence of number of vehicles on transmission delay.

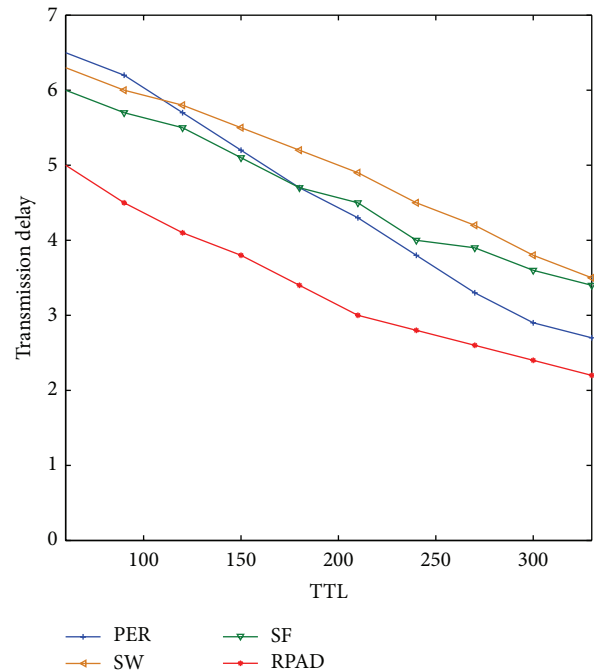


FIGURE 8: The influence of TTL on transmission delay.

transmission ratio and transmission delay. When the total number of vehicles increases, the transmission ratio increases and transmission delay decreases. In fact, when there are many vehicles in network, messages are transmitted from one vehicle to another quickly, and vehicles do not need to carry the messages for a long time. It can be seen from Figures 5 and 6 that RPAD is superior to SW, SF, and PER.

5.4.3. *Influence of Time to Live (TTL)*. Figures 7 and 8 demonstrate the influence of the total number of time to live values on transmission ratio and transmission delay. If the value of TTL is low, all algorithms perform terribly. When the value of TTL increases, transmission ratio increases, while transmission delay decreases, and when more and more nodes are dead, it will take longer for nodes to transmit

messages to others. It is clearly observed from Figures 7 and 8 that RPAD holds the higher transmission ratio and shorter transmission delay.

6. Conclusions

In this paper, we address the conflict of two vehicles happening to choose the same parking space in vehicular delay-tolerant network. We design the routing protocol RPAD from the direction and distribution density of the vehicle and the base station is considered as a reference. The vehicles having the same direction of that along the reference direction and higher distribution density are chosen for the next hop. A multiparty computation based privacy protection scheme is also proposed for security consideration.

The remaining work is the validations of our scheme on real data and applications in various delay-tolerant environments where generalizing the assumptions will be taken. Due to the limited time and conditions, now we do not have so many vehicles to solidify the proposed approach, and the demonstration of the validity through some real cases will be done in the further work.

Conflict of Interests

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

Acknowledgments

This work was supported in part by grants from the National Natural Science Foundation of China (nos. 61202355, 61201163, 61100199, and 61373138), the Natural Science Foundation of Jiangsu Province (no. BK20141429), Scientific and Technological Support Project (Industry) of Jiangsu Province (no. BE2013666), and Natural Science Key Fund for Colleges and Universities in Jiangsu Province (no. 12KJA520002).

References

- [1] R. X. Lu, X. D. Lin, T. H. Luan et al., "PreFilter: an efficient privacy-preserving relay filtering scheme for delay tolerant networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 1395–1403, Orlando, Fla, USA, March 2012.
- [2] J. Hur and K. Kang, "Secure data retrieval for decentralized disruption-tolerant military networks," *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 16–26, 2014.
- [3] X. Lv, Y. Mu, and H. Li, "Non-interactive key establishment for bundle security protocol of space DTNs," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp. 5–13, 2014.
- [4] S. Rane, W. Sun, and A. Vetro, "Secure distortion computation among untrusting parties using homomorphic encryption," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '09)*, pp. 1485–1488, Cairo, Egypt, November 2009.
- [5] S. Rane, W. Sun, and A. Vetro, "Privacy-preserving approximation of L1 distance for multimedia applications," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '10)*, pp. 492–497, IEEE, Singapore, July 2010.
- [6] Y.-G. Yang, J. Xia, X. Jia, and H. Zhang, "Comment on quantum private comparison protocols with a semi-honest third party," *Quantum Information Processing*, vol. 12, no. 2, pp. 877–885, 2013.
- [7] H.-S. Huang, H. Zhong, F.-F. Yan, and Y.-F. Sun, "Two protocols for no-information leaked closest-pair of points," *Computer Engineering and Applications*, vol. 46, no. 34, pp. 80–81, 2010.
- [8] L. Gao, M. Li, M. Zhou, and W. Shi, "Privacy protected data forwarding in human associated delay tolerant networks," in *Proceedings of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom '13)*, pp. 586–593, Melbourne, Australia, July 2013.
- [9] H. Samuel, W. Zhuang, and B. Preiss, "Improving the dominating-set routing over delay-tolerant mobile ad-hoc networks via estimating node intermeeting times," *Eurasip Journal on Wireless Communications and Networking*, vol. 2011, Article ID 402989, 12 pages, 2011.
- [10] F. Xu, S. Guo, J. Jeong et al., "Utilizing shared vehicle trajectories for data forwarding in vehicular networks," in *Proceedings of the 30th IEEE International Conference on Computer Communications*, pp. 441–445, Kyoto, Japan, 2011.
- [11] C. Dunbar and G. Qu, "A DTN routing protocol for vehicle location information protection," in *Proceedings of the IEEE Military Communications Conference (MILCOM '14)*, pp. 94–100, Baltimore, Md, USA, October 2014.
- [12] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: social-based forwarding in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576–1589, 2011.
- [13] F. Fabbri and R. Verdone, "A sociability-based routing scheme for delay-tolerant networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, Article ID 251408, 2011.
- [14] C.-J. Lee, S.-Y. Kang, and K.-I. Kim, "Design of hierarchical routing protocol for heterogeneous airborne ad hoc networks," in *Proceedings of the 28th International Conference on Information Networking (ICOIN '14)*, pp. 154–159, IEEE, Phuket, Thailand, February 2014.
- [15] Y. Liu, J. Niu, G. Qu, Q. Cai, and J. Ma, "Message delivery delay analysis in VANETs with a bidirectional traffic model," in *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC '11)*, pp. 1754–1759, Istanbul, Turkey, July 2011.

Research Article

Sensor Networks Hierarchical Optimization Model for Security Monitoring in High-Speed Railway Transport Hub

Zhengyu Xie^{1,2} and Yong Qin²

¹School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

²State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Yong Qin; yqin@bjtu.edu.cn

Received 21 November 2014; Revised 31 March 2015; Accepted 7 April 2015

Academic Editor: Fei Yu

Copyright © 2015 Z. Xie and Y. Qin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We consider the sensor networks hierarchical optimization problem in high-speed railway transport hub (HRTH). The sensor networks are optimized from three hierarchies which are key area sensors optimization, passenger line sensors optimization, and whole area sensors optimization. Case study on a specific HRTH in China showed that the hierarchical optimization method is effective to optimize the sensor networks for security monitoring in HRTH.

1. Introduction

With the rapid development of high-speed railway in China, many modern HRTHs have been built to match the developing demands. HRTHs become the crossing and interface of multitransportation which include high-speed railway, civil aviation, highway, waterway, urban rail transit, public transport, motor vehicle, and taxi. As a vital node of passenger transport net, HRTH is an important distribution place of massive passenger flow. With the increase of high-speed railway operation mileage, the distribution quantity of passengers will be sustained to increase sharply, which leads HRTHs to confront severe challenges in passenger flow security monitoring.

At present, the video surveillance system is the main security monitoring approach used in HRTH. The managers can detect the congestion of passenger flow, abnormal behaviors of passengers, abandoned objects, and so forth by using surveillance systems. The basic workflow of system includes the following: (i) data acquisition: distribute surveillance sensors and develop a sensor network; (ii) data transmission: choose suitable approaches to transmit data acquired from the sensor networks; (iii) data processing: utilize efficient image processing method to process the data acquired from the sensor networks and obtain processing result based on the demands of security monitoring; (iv) data dissemination:

select various channels to disseminate the security monitoring information. Currently, the studies related to video surveillance system in HRTH mainly focused on (ii) and (iii) to improve detection accuracy and speed; specific study on (i) is scarce. As a foundation of other parts, the sensor networks have important influences on other parts. So it is necessary for HRTH security monitoring to optimize the sensor networks.

The rest of this paper is organized as follows: The relevant literature is reviewed in the next section. The sensor networks hierarchical optimization problem is described in Section 3 and Section 4 proposes a sensor networks hierarchical optimization model. A case study is reported in Section 5 and finally Section 6 covers the conclusion.

2. Literature Review

The sensor networks optimization problem for security monitoring in HRTH belongs to the art gallery problem (AGP) which was first proposed in 1973 in a conversation between Klee and Chvatal [1]. Based on the conversation, Chvatal proved $[n/3]$ cameras are always sufficient and sometimes necessary. This conclusion is called the Art Gallery Theorem, or Watchman Theorem [2]. Fisk used triangulation techniques and staining methods and got the conclusion “any simple polygon after triangulation, the corresponding diagram can 3- stain,” and the same type of colored dots

TABLE 1: Data acquisition demands of security monitoring in HRTH.

Level	Description	Concern	Data acquisition demands
First level	Key area monitoring	Focus on the security of key, important, and sensitive areas	The data of key areas in HRTH must be continuously acquired and can meet the anomaly detection of key areas
Second level	Passenger line monitoring	Focus on the security of passenger input and output lines	The data of entire passenger line in HRTH must be continuously acquired and can meet the forecast demands of post node in passenger line
Third level	Complete coverage monitoring	Focus on the security of whole HRTH	The data of whole HRTH can be inconsecutively and optionally acquired and must ensure all function areas are completely covered

can cover the entire simple polygon [3]. Avis, Toussaint, and Chazelle gave different algorithms for the simple polygon triangulation. For any simple polygon with given point, we can determine the location of monitors in a simple polygon within time, making any point in this simple polygon able to see at least one monitor [4, 5]. Lee and Lin proved that the algorithm of solving any simple polygon which required minimum number of guards is NP-hard [6].

After Art Gallery Theorem is proved, more and more questions of AGP are proposed, including the following: the monitor can be moved at the edge, the monitor can be moved at the diagonal, at least two monitors are required that can be guarded by each other, one guard is removed while the other guards could know, and the walls of the gallery should be vertical [7, 8].

In computational geometry, the gallery can be abstract simple polygon; put a monitor abstraction for a point in simple polygon, and then the problem can be abstracted as an art gallery plane geometry problem; gallery guards problem can be abstracted as how many points can cover the entire simple polygon. Variant problem can be abstracted as joint guards, side cover, diagonal coverage, coguards, orthogonal gallery guards, moving guard, limited perspective guards, moving guard with limited perspective, orthogonal polygons mobile guards, and other issues [9, 10].

For unrealistic assumptions of monitor in the solving of AGP and its variant problem, such as magnifying the monitoring range of single monitor, expanding the depth of field, and not limiting the recognition accuracy and speed, lead to research in art galleries and related issues are hard to be good application in the actual layout of video surveillance capture point.

Applied researches of monitor sensors layout mainly put video monitor sensors layout problem into set covering problem. Chakrabarty and Bulusu used the method of linear programming to obtain the minimum activity to maintain coverage node set [11, 12]. Meguerdichian et al. made more complex coverage model which, from the perspective of minimizing the uncovered area of the start, considers the problem of network coverage uniformity runtime based on the degree of coverage [13]. Erdem and Sclaroff proposed an efficient algorithm to calculate the radial scans of each collection point in the visual range of the camera, so that the total layout costs are optimized while the collection point layout constraints can be met [14].

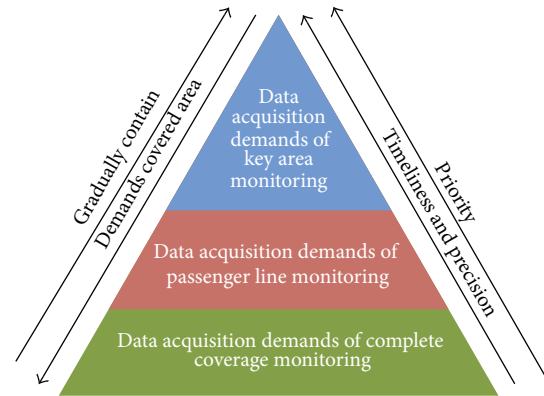


FIGURE 1: Demands relationship among three levels.

3. Problem Description

In this section, the sensor networks hierarchical optimization problem is described in three aspects. Firstly, the data acquisition demands of security monitoring in HRTH are analyzed. Secondly, the hierarchical organization of sensor networks is described. Based on the previous two parts, the basic process of security monitoring based on multilayer sensor networks is designed in the last part.

3.1. Data Acquisition Demands of Security Monitoring in HRTH. According to the different safety forewarning focuses, the security monitoring can be divided into three levels, and each level has its specific data acquisition demands. The data acquisition demands of security monitoring in HRTH are shown in Table 1.

The demands relationship among three levels is shown in Figure 1. The demands covered areas are gradually increasing from the first level to the third level, and the timeliness and precision of data acquisition are gradually increasing from the opposite direction.

3.2. Hierarchical Organization of Sensor Networks. Based on the data acquisition demands analysis above, the sensor networks for security monitoring in HRTH are classified into three hierarchies, which are one-to-one correspondence to the data acquisition demands levels. The hierarchical organization of sensor networks is shown in Table 2.

TABLE 2: Hierarchical organization of sensor networks.

Sensor network	First hierarchy	Key area monitoring sensors	(i) Sensors in different key areas are independent and do not have any relevance (ii) Sensors do not need adjustment after setting (iii) Sensors have front-end event detecting software
	Second hierarchy	Passenger line monitoring sensors	(i) Sensors should be set following the passenger line (ii) Sensors in same passenger line have association (iii) Sensors do not need adjustment after setting (iv) The data acquired by sensors should be continuously transferred to the control center to process
	Third hierarchy	Complete coverage monitoring sensors	(i) Sensors should cover all the function areas in HRTH (ii) The monitoring areas of sensors should reduce overlaps as much as possible (iii) Sensors can adjust monitoring areas after setting (iv) The data acquired by sensors should be continuously transferred to the control center to be stored

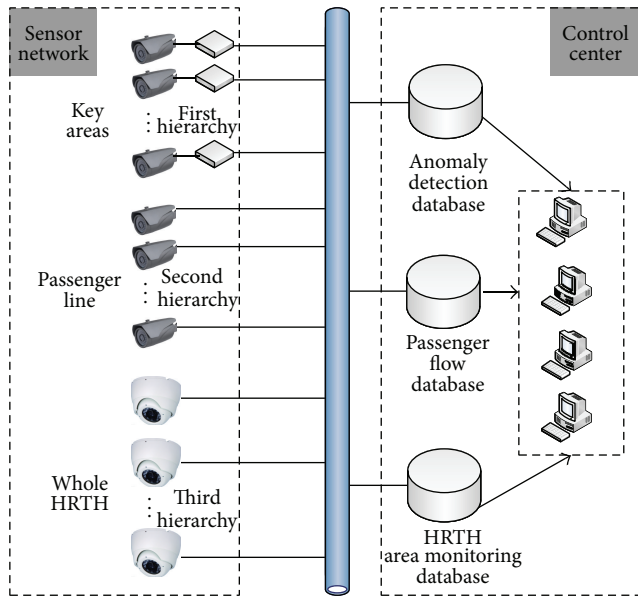


FIGURE 2: Basic structure of security monitoring.

3.3. Basic Structure of Security Monitoring Based on Multilayer Sensor Networks. According to the above analysis in this section, a basic process of security monitoring is designed based on multilayer sensor networks. The structure is shown in Figure 2.

As observed in Figure 2, in the first hierarchy, anomalies in key areas are detected by monitoring sensors, and then the anomaly detection data are transmitted to control center and inform monitoring personnel to respond. In the second hierarchy, passenger flow data are acquired by monitoring sensors and transmitted to control center. According to the incidence relation among sensors, the passenger flow data can be processed to obtain real-time passenger flow status, search the post node in passenger line, and forecast the variation trend of passenger flow. The monitoring personnel can forewarn passenger flow congestion and make emergency

response based on the processing results. The third hierarchy mainly focuses on the overall safe state of HRTH. The monitoring personnel need to use the monitoring sensors to observe function areas in HRTH when the first or second hierarchy has safety forewarning. This hierarchy is a supplement for the previous two hierarchies.

4. Sensor Networks Hierarchical Optimization Model

According to the problem description in Section 3, a sensor networks hierarchical optimization model is proposed in this section. Sensor networks for security monitoring in HRTH are optimized from three hierarchies based on the hierarchical organization mentioned above. The hierarchical optimization framework is shown in Figure 3.

4.1. Key Area Sensors Optimization. The core concern of key area sensors optimization is to determine the key areas in HRTH. According to the different area characteristic, the key areas can be mainly divided into congestion areas and sensitive areas. Each area has its specific determination method.

4.1.1. Congestion Areas Determination Method. The congestion areas are mainly determined by the computation result. There are three main methods to calculate the relation between passenger flow and facilities capacity, which are described as follows.

(1) Capacity Method. Capacity method is used to determine facilities congestion. This method divides the passenger line into several units and calculates the capacity balance of units. When the facility design capacity is less than practical capacity, this facility is considered a key area. The facility design capacity is calculated by

$$C = W \cdot q \cdot \varphi. \quad (1)$$

W is the width of the facility. q is the predicted passenger flow volume. φ is the peak period coefficient.

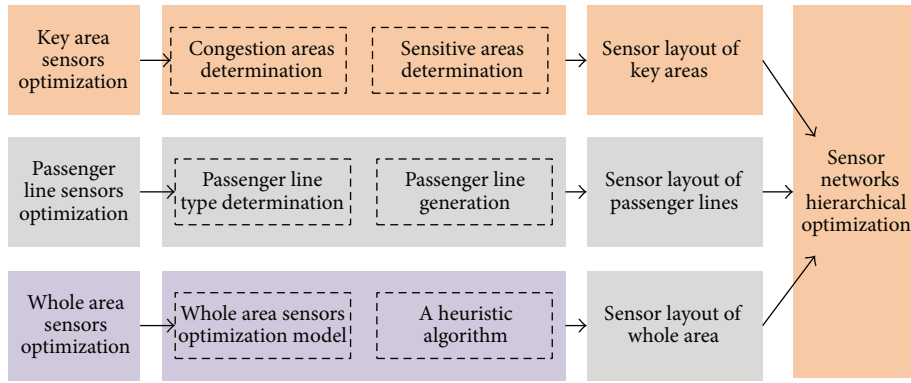


FIGURE 3: Hierarchical optimization framework.

TABLE 3: Calculation of three behaviors.

Delay behaviors	Generation mechanism	Calculation	Range of application
Queuing delay	The facility service capability is less than passenger arrival rate	$\frac{\lambda}{(\mu - \lambda)^2}$	Ticket entrance, wicket, baggage check entrance, and so forth
Congesting delay	The facility cannot be used when the passengers arrive	$\int_{t_0}^{t_n} q(t) dt - n \int_{nT+(n-1)q(t)}^{nT+(n-1)(k_1+k_2)q(t)} q(t) dt$	Ticket entrances delay check caused by train late
Waiting delay	The facility capacity is insufficient, which leads to high density and low speed of passenger flow	$\frac{Lk_j}{ku_f}$	The service capability of interface channel between service nodes is insufficient

(2) *Delay Method*. Delay is an important judging parameter for the congestion of passenger line. The passenger delay in HRTH mainly results from queuing, congesting, and waiting behaviors. The calculation of three behaviors is shown in Table 3.

(3) *Density Method*. Passenger flow density is an effective indicator to measure congestion level. The higher density passenger flow has, the more congestion in passenger line arises. This density is named congestion density and calculated by

$$\rho_{ij}(t) = \frac{n_{ij}(t)}{M_{ij}}. \quad (2)$$

$\rho_{ij}(t)$ is congestion density of the j th segment in the i th passenger line. M_{ij} is the facility available area of the j th segment in the i th passenger line. $n_{ij}(t)$ is passenger amount of the j th segment in the i th passenger line.

4.1.2. Sensitive Areas Determination Method. Sensitive areas determination, compared with congestion areas determination, is relatively simple and does not have specific calculating method. Most of sensitive areas are determined based on the actual demand of security monitoring in HRTH. The common sensitive areas include distribution facility areas, fireproofing facility areas, office areas, and security check areas.

4.2. Passenger Line Sensors Optimization. The core concerns of passenger line sensors optimization are to determine the passenger line type and generate the passenger line under established facility layout.

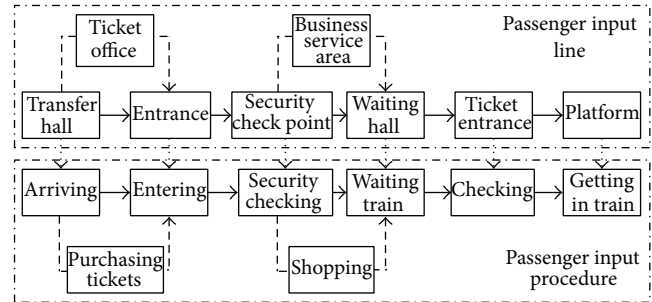


FIGURE 4: Passenger input line and procedure.

4.2.1. Passenger Line Type Determination. The passenger line in HRTH can be mainly divided into passenger output line, passenger input line, and passenger transfer line. These three types of passenger lines are described as follows.

(1) *Passenger Input Line*. The passenger input line begins at passenger arriving at HRTH and finishes after passenger gets in trains. In the period between passenger arriving and leaving, there are several events happening, such as purchasing tickets, shopping, dining, and waiting in trains. The passenger input line and procedure are shown in Figure 4.

(2) *Passenger Output Line*. The passenger output line begins at train arriving at HRTH and finishes after passenger leaves HRTH. Compared with passenger input line, passenger output line has few events and is relatively simple. The output passengers flow has characteristics of being concentrated,

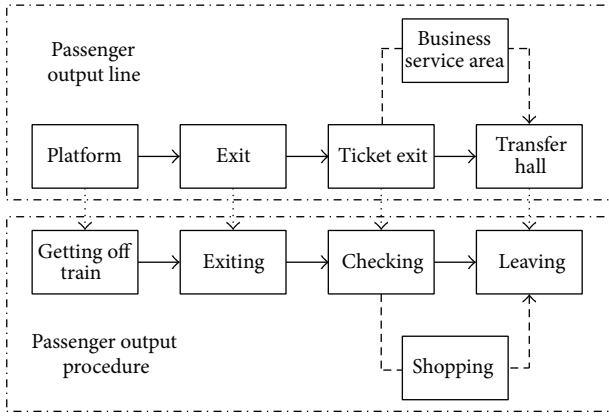


FIGURE 5: Passenger output line and procedure.

high density, and short stay time. The passenger output line and procedure are shown in Figure 5.

(3) *Passenger Transfer Line*. Passenger transfer line is similar to passenger input line and relatively simple, so we do not introduce it in detail.

4.2.2. Passenger Line Generation. After determining the passenger line type, passenger line is generated based on the established facility layout of HRTH. Generation steps are shown as follows.

Step 1. Mark the geometric center of facilities in HRTH function areas and use these geometric centers as the origin-destination points.

Step 2. Use directed line segments to link geometric centers based on passenger moving tracks in different type passenger lines.

Step 3. Classify the directed line segments, use different color to denote different type passenger lines, and use different thicknesses lines to denote the amount of passenger flow.

4.3. Whole Area Sensors Optimization. In order to ensure that all function areas in HRTH are covered by sensors, a whole area sensors optimization framework is proposed in this section. After space two-dimension, space partition and visibility analysis, we change the whole area sensors optimization problem into set covering problem and develop a set covering model. The whole area sensors optimization framework is shown in Figure 6.

Step 1 (HRTH space two-dimension). HRTH space two-dimension is to make the three-dimensional space into a two-dimension ichnography and scale down the layout of facilities and instruments. After HRTH space two-dimension, we can obtain a schematic representation of whole HRTH.

Step 2 (HRTH space partition). After obtaining the schematic representation, we abstract the facilities and instruments into square or rectangle and lengthen the sides of

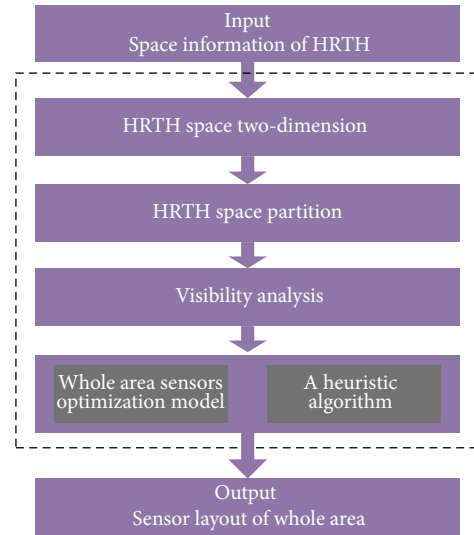


FIGURE 6: Whole area sensors optimization framework.

square and rectangle. A space partition sample is shown in Figure 7.

Step 3 (visibility analysis). Based on HRTH space partition, we analyze the visibility of each region in schematic representation. The visibility analysis includes two parts.

The first part is to analyze the geometric visibility. Assume the region center is the laying position. If the link line between two regions is not interrupted by facilities or instruments, the two regions are considered geometric visibility. Figure 8(a) is a geometric visibility analysis sample. The geometric visibility set of R_{12} is $\{R_3, R_4, R_5, R_7, R_8, R_9, R_{10}, R_{11}, R_{13}, R_{14}, R_{15}, R_{16}, R_{17}\}$.

The second part is visual range analysis. We set the coverage of one sensor as a circle whose radius is the visual range of sensor. The regions which are covered by the circle are the visual regions. Figure 8(b) is a visual range analysis sample. The visual visibility set of R_{12} is $V_{12} = \{R_7, R_8, R_{10}, R_{11}, R_{13}, R_{14}, R_{17}\}$.

Step 4 (a set covering model for whole area sensors optimization). In this step, we develop a set covering model to describe the whole area sensors optimization problem.

(1) *Notations and Variables.* Consider the following:

i, j : region index,

m : total amount of regions,

R : visual range of sensor,

$\text{dis}(i, j)$: the distance between region i and region j ,

W_i : weighted values,

x_i : 0-1 variable; if sensor is set in region i , $x_i = 1$; otherwise, $x_i = 0$,

$\text{cov}(i, j)$: 0-1 variable; if the visual range of region i can cover region j , $\text{cov}(i, j) = 1$; otherwise, $\text{cov}(i, j) = 0$.

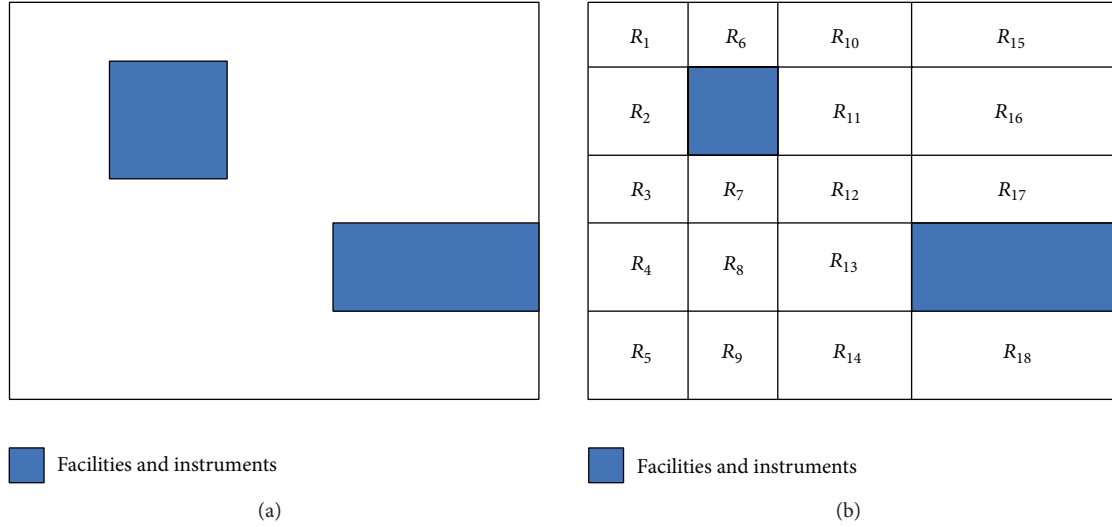


FIGURE 7: A space partition sample.

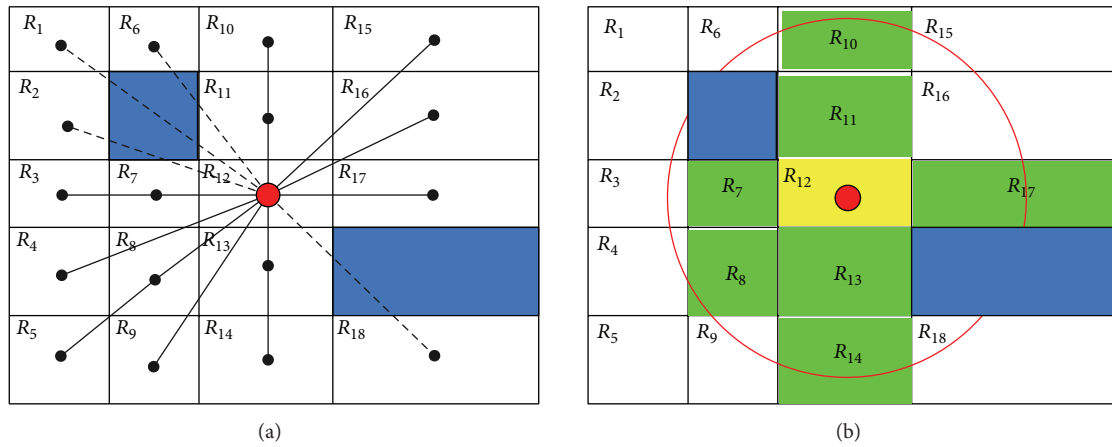


FIGURE 8: Visibility analysis samples.

(2) *Objective Function.* The objective function of whole area sensors optimization model is written as follows:

$$\text{Minimize } Z = \sum_{i=0}^m W_i x_i. \quad (3)$$

The objective function minimizes the amount of sensors to cover whole function areas in HRTH.

(3) *Constraints.* The constraints of whole area sensors optimization model are introduced as follows to ensure the practical feasibility of the solution:

$$\sum_{i_1=0}^m \text{cov}(i, j) x_{i_1} \geq 1, \quad 0 \leq j \leq m, \quad (4)$$

$$\text{cov}(i, j) \cdot (R - \text{dis}(i, j)) \geq 0, \quad (5)$$

$$0 \leq i \leq m, \quad 0 \leq j \leq m,$$

$$(1 - \text{cov}(i, j)) \cdot (\text{dis}(i, j) - R) \geq 0, \quad (6)$$

$$0 \leq i \leq m, \quad 0 \leq j \leq m,$$

$$x_i \in \{0, 1\}, \quad (7)$$

$$\text{cov}(i, j) \in \{0, 1\}. \quad (8)$$

Constraint (4) represents that each region in HRTH should at least be covered by one sensor. Constraint (5) ensures that the distance between sensor and covered region cannot be larger than the visual range of sensor. Constraint (6) represents that the region whose distance is larger than the visual range of sensor cannot be covered by this sensor. Constraint (7) and Constraint (8) are 0-1 variable constraints.

Step 5 (solution algorithm). In order to solve the optimization model developed above, a heuristic algorithm is proposed in this section. X is the area set of two-dimension division. m is the elements amount in X . C is the area set of

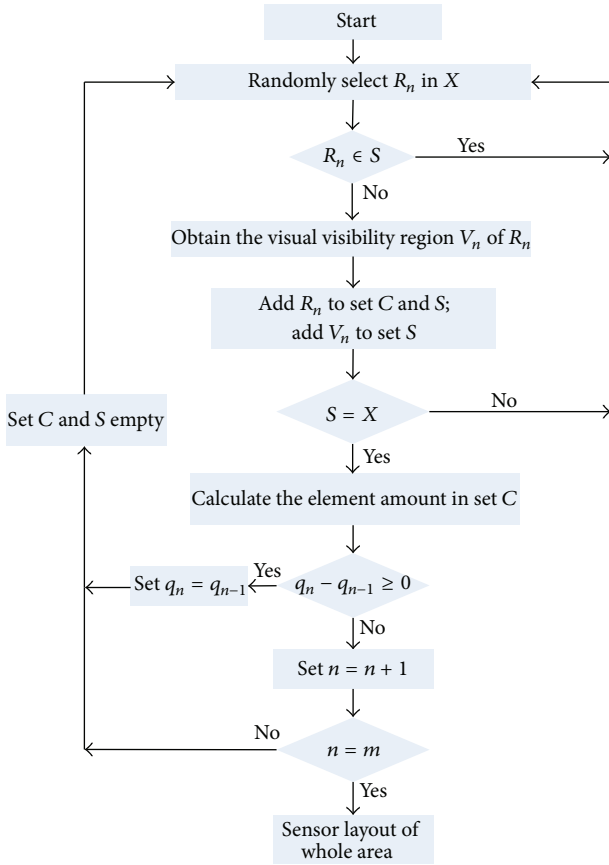


FIGURE 9: Algorithm implement process.

sensors layout. q is the elements amount in C . The algorithm implement process is shown in Figure 9.

5. A Case Study

To illustrate the proposed model and algorithm for sensor networks hierarchical optimization problem, a case study is performed by using the actual data from a specific HRTH in China. We choose comprehensive transfer layer of the HRTH as optimization object.

This layer is composed by transfer hall, parking area, passenger output system, and passenger input system. There are six entrances, six exits, and four ticket offices in this layer. The transfer hall connects with metro, taxi, and bus. The whole layer has various kinds of passenger lines and crossover among passenger lines.

We use the hierarchical optimization method mentioned in Section 4 to optimize the sensor networks in this layer. The hierarchical optimization is shown as follows.

5.1. Key Area Sensors Optimization. According to the actual passenger flow data in this layer, we use the key area determination method mentioned in Section 4.1 to determine key areas. The distribution of key areas in this layer is shown in Figure 10.

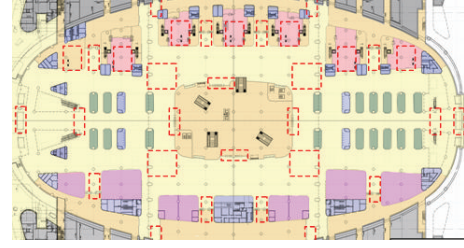


FIGURE 10: Distribution of key areas.



FIGURE 11: Passenger lines in the layer.

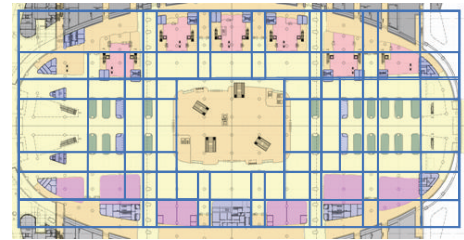


FIGURE 12: Space partition process.

5.2. Passenger Line Sensors Optimization. Through analysis of origin-destination points and passenger moving tracks, we generate the passenger lines in this layer. The passenger lines are shown in Figure 11.

5.3. Whole Area Sensors Optimization. Follow the whole area sensors optimization framework mentioned in Figure 6; the space is partitioned which is shown in Figure 12 and the space partition result is shown in Figure 13.

After space partition, the layer is divided into 58 regions. Through the visibility analysis, we can obtain the visual visibility sets of 58 regions and use the heuristic algorithm to find a solution for the whole area sensors optimization model. The final sensors layout region set is $\{R_2, R_5, R_6, R_9, R_{12}, R_{14}, R_{21}, R_{22}, R_{29}, R_{30}, R_{34}, R_{36}, R_{41}, R_{49}, R_{50}, R_{57}\}$. The solution obtained by the heuristic algorithm is shown in Figure 14.

According to the hierarchical optimization, the final sensor networks for security monitoring in HRTH are shown in Figure 15.

6. Conclusion

In this paper, we considered the sensor networks hierarchical optimization problem in HRTH. A hierarchical optimization framework is proposed, and the problem is solved from

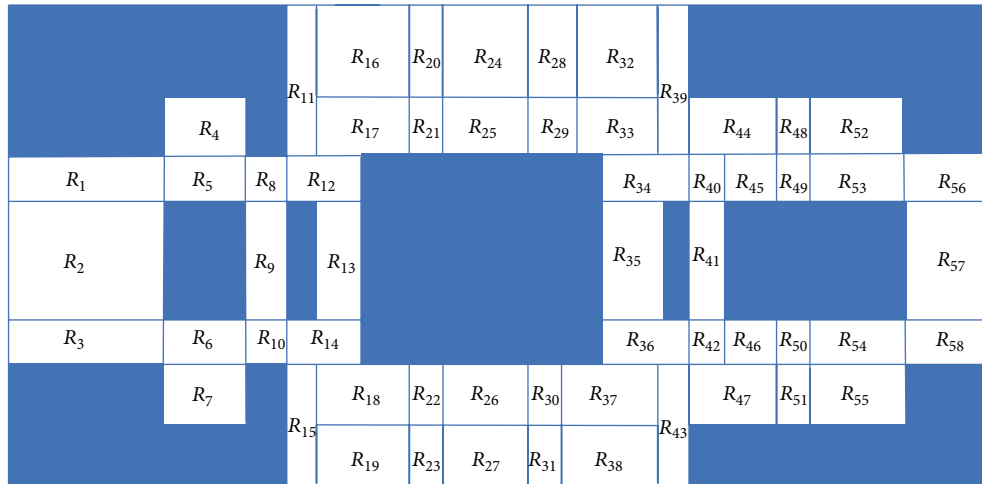


FIGURE 13: Space partition result.

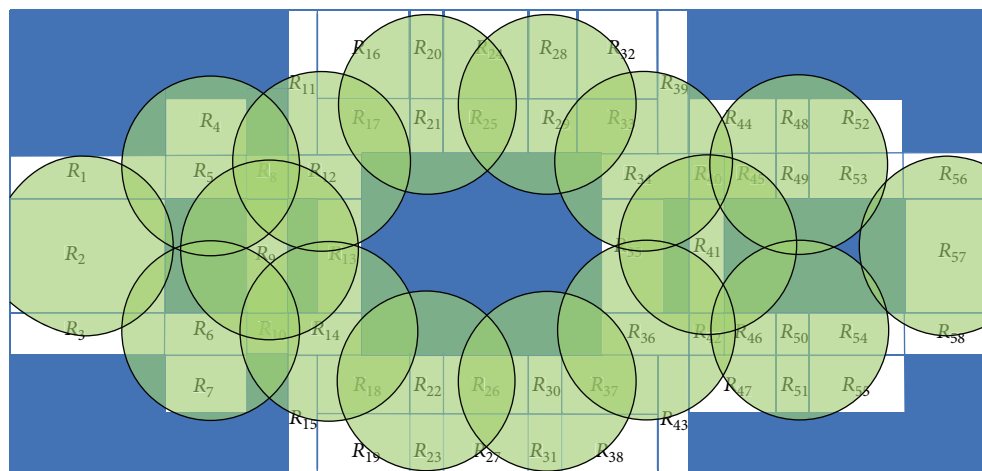
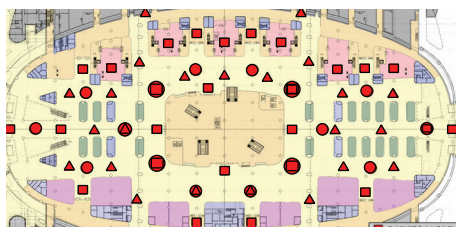


FIGURE 14: Solution obtained by the heuristic algorithm.



- Sensors for key areas
- ▲ Sensors for passenger lines
- Sensors for whole areas

FIGURE 15: Final sensor networks for security monitoring in HRTH.

three hierarchies which are key area sensors optimization, passenger line sensors optimization, and whole area sensors optimization. In the third hierarchy, a whole area sensors

optimization model is developed and a heuristic algorithm is designed. Case study on a specific HRTH in China showed that the hierarchical optimization method is effective to optimize the sensor networks for security monitoring in HRTH. In the future, considering the layout costing in optimization method is a possibility for further research.

Conflict of Interests

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

Acknowledgments

This research was supported by the Fundamental Research Funds for the Central Universities (Grant no. 2015JBM044), the National Natural Science Foundation of China (Grant no. 61374157), and the Talented Faculty Funds of Beijing Jiaotong University (Grant no. 2014RC005).

References

- [1] A. Aggarwal, *The art gallery theorem: its variations, applications and algorithmic aspects [Ph.D. thesis]*, Johns Hopkins University, Baltimore, Md, USA, 1984.
- [2] V. Chvatal, "A combinatorial theorem in plane geometry," *Journal of Combinatorial Theory Series B*, vol. 18, pp. 39–41, 1975.
- [3] S. Fisk, "A short proof of Chvatal's watchmen theorem," *Journal of Combinatorial Theory Series B*, vol. 24, no. 3, 374 pages, 1978.
- [4] D. Avis and G. T. Toussaint, "An efficient algorithm for decomposing a polygon into star-shaped polygons," *The Journal of the Pattern Recognition Society*, vol. 13, no. 6, pp. 395–398, 1981.
- [5] B. Chazelle, "A theorem on polygon cutting with applications," in *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science (SFCS '82)*, pp. 339–349, Chicago, Ill, USA, November 1982.
- [6] D. T. Lee and A. K. Lin, "Computational complexity of art gallery problems," *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 276–282, 1986.
- [7] J.-I. Doh and K.-Y. Chwa, "An algorithm for determining visibility of a simple polygon from an internal line segment," *Journal of Algorithms*, vol. 14, no. 1, pp. 139–168, 1993.
- [8] Y. Ke, "Detecting the weak visibility of a simple polygon and related problems," Tech. Rep., Johns Hopkins University, Baltimore, Md, USA, 1987.
- [9] J. O. Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, New York, NY, USA, 1987.
- [10] T. C. Shermer, "Recent results in art galleries," *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1384–1399, 1992.
- [11] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Grid coverage for surveillance and target location in distributed sensor networks," *IEEE Transactions on Computers*, vol. 51, no. 12, pp. 1448–1453, 2002.
- [12] N. Bulusu, J. Heidemann, and D. Estrin, "Adaptive beacon placement," in *Proceedings of the 21st IEEE International Conference on Distributed Computing Systems*, pp. 489–498, April 2001.
- [13] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless ad-hoc sensor networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pp. 139–150, July 2001.
- [14] U. M. Erdem and S. Sclaroff, "Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements," *Computer Vision and Image Understanding*, vol. 103, no. 3, pp. 156–169, 2006.

Research Article

Intrusion Detection Algorithm for Mitigating Sinkhole Attack on LEACH Protocol in Wireless Sensor Networks

Ranjeeth Kumar Sundararajan and Umamakeswari Arumugam

School of Computing, SASTRA University, Tirumalaisamudram, Thanjavur, Tamil Nadu 613401, India

Correspondence should be addressed to Ranjeeth Kumar Sundararajan; ranjeethkumar@sastra.edu

Received 15 October 2014; Revised 5 January 2015; Accepted 12 February 2015

Academic Editor: Fei Yu

Copyright © 2015 R. K. Sundararajan and U. Arumugam. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In wireless sensor network (WSN), the sensors are deployed and placed uniformly to transmit the sensed data to a centralized station periodically. So, the major threat of the WSN network layer is sinkhole attack and it is still being a challenging issue on the sensor networks, where the malicious node attracts the packets from the other normal sensor nodes and drops the packets. Thus, this paper proposes an Intrusion Detection System (IDS) mechanism to detect the intruder in the network which uses Low Energy Adaptive Clustering Hierarchy (LEACH) protocol for its routing operation. In the proposed algorithm, the detection metrics, such as number of packets transmitted and received, are used to compute the intrusion ratio (IR) by the IDS agent. The computed numeric or nonnumeric value represents the normal or malicious activity. As and when the sinkhole attack is captured, the IDS agent alerts the network to stop the data transmission. Thus, it can be a resilient to the vulnerable attack of sinkhole. Above all, the simulation result is shown for the proposed algorithm which is proven to be efficient compared with the existing work, namely, MS-LEACH, in terms of minimum computational complexity and low energy consumption. Moreover, the algorithm was numerically analyzed using TETCOS NETSIM.

1. Introduction

Wireless sensor devices are used in a broad range of applications such as defense, farming, medicine, and industries. WSNs deploy an array of microsensors that senses the activities of a physical phenomenon and sends the information to the base station (BS). They face a lot of security issues that arise due to their low operating energy and minimal computational capabilities. Table 1 shows some of the security attacks in the different layers of the WSN protocol stack. This research work focuses on the network layer threat and its effects, namely, sinkhole attack which is set off by a malicious node that attracts the traffic from its neighboring nodes and either selectively forwards it or alters it, resulting in a successful intrusion and high data loss rate of the real time data [1]. An extensive study on sensor routing protocols and their attacks like selective forwarding, spoofing/replaying, sinkhole, wormhole, Sybil attack, and HELLO flood attack with the counter actions is available in [2]. Various security threats in WSN are identified and classified into two broad

categories, namely, active and passive attacks. In active attack, the compromised node alters or makes changes in the data during transmission. Some of the active attacks are denial of service (DoS), modification, impersonation, fabrication, and so on. In passive attack, the malicious node does not make any changes in the data, but it overhears the data transmission. Some of the passive attacks are eavesdropping, traffic analysis, and camouflage adversaries.

The security mechanisms to counter these attacks are classified into two types, namely, low level and high level. The low level mechanism includes key establishment, privacy, and authentication. The high level mechanism includes secure group management, Intrusion Detection System (IDS), and secure data aggregation [3]. The IDS forms a second level of defense to the network and alerts the network in the presence of threats. There are four different types of IDS, namely, Signature IDS, Anomaly IDS, Hybrid IDS, and Cross Layer IDS. These IDS can be compared based on the characteristics like detection rate, false alarm rate, computational capability, energy consumption rate, and so on [4, 5]. One of

TABLE 1: Layer-wise threats.

Layer	Threats
Physical	Jamming, tampering
Data link	Collision, exhaustion, unfairness
Network	Sinkhole, black hole, selective forwarding
Transport	Flooding, false messages, desynchronization
Application	Reliability attack, clock skewing, data aggregation distortion

the important low level security mechanisms is cryptographic method, which includes key size, block size, and message about the round as corresponding information. Many security protocols like TinySec, MinSec, SPINS, and LSec are proposed to provide security to the sensor network and these protocols use encryption and authentication mechanisms [6].

This paper focuses on the high level defense mechanism, namely, IDS, to detect the malicious nodes. The malicious node launches the attack by advertising that it is the nearest node to the BS and attracts the packets and alters those passing through it. It still remains an open weakness in case of insider attacks, where a node is free to manipulate the packets and gain control over it. Most routing protocols in the sensor network do not initiate any mechanisms for detecting security attacks. Encryption methodologies and authentication system prove to be inefficient in the case of laptop and insider attacks. So, it has become imperative to devise a mechanism against these attacks practically. The main objective of this research work is to study the effects of sinkhole attack in a WSN which uses the LEACH protocol for its routing operation and devise a security mechanism to overcome the adverse effects. Sinkholes are induced in a WSN either by insiders or by an external attacker. The proposed IDS algorithm detects the sinkhole attack with high detection rate. The performance of the intrusion detection algorithm is verified numerically and simulations enforce the accuracy and the effectiveness of the algorithm. Four main contributions in this work are as follows.

- (1) A lightweight IDS is proposed with minimal computational complexity.
- (2) A novel intrusion detection metric, namely, intrusion ratio (IR), is introduced.
- (3) A detailed security analysis is performed in three different scenarios.
- (4) The proposed lightweight IDS is capable of capturing multiple attacks.

This paper is structured as follows: Section 2 gives the similar IDS works on sinkhole attacks and LEACH protocol. The description of LEACH protocol, sinkhole attack, and research motivation is presented in Section 3. Section 4 includes methods of launching the sinkhole attack in LEACH and the proposed IDS algorithm. Section 5 shows the simulation of sinkhole attack and analysis of proposed algorithm. In Section 6, conclusion and future work are given.

2. Related Work

An efficient IDS algorithm with low overhead was proposed by Ngai et al. [1]. This robust algorithm checks the data consistency and captures the intruder by verifying the network flow information. The algorithm is also robust against the presence of multiple malicious nodes. In [7], different ways to launch the sinkhole attack are discussed. The BS is identified as the trusted member in the network. Based on the sequence number, the sinkhole attack was launched and subsequently the packet transmission was performed through the Ad Hoc On-Demand Distance Vector (AODV) protocol to identify the malicious activity of the intruder. The authors in [8] propose a two-step intrusion detection process to detect the colluding nodes acting against the BS. They analyze the network routing patterns for data consistency. Based on the node ID's the BS identifies the compromised node and alerts the normal sensor nodes. The impact of wormhole attack on LEACH protocol has been analyzed [9]. A separate tunnel is created by the attacker through which the data is transferred to the wormhole nodes. The wormhole attack can also be used to launch the sinkhole attack by making one of the wormhole nodes a sinkhole. An IDS to detect the sinkhole attack in the WSN which uses Mintroute protocol for its routing operation was proposed [10]. Using a strategy of advertisement, the sinkhole attack was launched that exploits the link quality of the compromised node to send the data to the sinkhole node. Thus, an IDS mechanism is developed as a localized agent to detect such malicious activity of the sinkhole node in the distributed networks.

The two security threats, namely, black hole and sinkhole attacks, are analyzed on the LEACH protocol [11]. The attacks are simulated in MATLAB with various metrics like residual energy, data transmission, and node longevity. The analyses were made in two different scenarios; normal operation and under attack. In [12], the proposed IDS integrate node behavior strategies and evidence theory. The multidimensional behavior characteristics are collected to calculate its deviation with the expected value and the belief factor is calculated for each sensor node. If the value of a sensor node is less than 0.25, it is blacklisted and marked as a malicious node. In [13], the proposed work has two approaches to detect sinkhole node in the network. The first approach identifies the region of the network which may contain the intruders. This work is performed in two ways by using geostatistical frailty survival model and distributed monitoring. The second approach is of mitigation type and this method is used to identify the intruders from the affected region. The authors in [14] propose IDS to detect the sinkhole attack. The sinkhole attack is launched on the Mintroute protocol by advertising a better link quality and changing the link quality value of current parent node to the worst value. They propose rule-based IDS to detect the sinkhole node. The authors also analyze the selective forwarding and black hole attacks on the Mintroute protocol. They also developed IDS to detect the attacks. In [15], the authors propose IDS to capture the sinkhole node which set itself as a fake BS. The node sends a control packet directly to the BS; then it sends data packet hop-by-hop. When the packet arrives, the IDS compares some of

its control fields with the original control packet and if any changes have been made to the control fields, then the IDS alert the presence of malicious node.

An IDS agent on each sensor node has two intrusion detection modules, namely, local agent and global agent [16]. The local agent stores the information of the sensor node, while the global agent monitors the communication of its neighbor nodes. The global agent uses watchdog and predefined 2-hop neighbor knowledge to detect the anomalies within its transmission range. In [17], the authors propose decentralized IDS which have watchdog modules residing in the monitor nodes. These nodes analyze the behavior of other nodes including cluster head (CH). If they detect an attack, the monitor node forwards the alarm to the BS and adds the compromised node to the blacklist. The blacklisted nodes are also broadcasted to all other nodes to avoid further communication. The authors in [18] propose IDS to identify the malicious activities according to the various phases of LEACH protocol. The CH selection is done according to the energy level for each round. When the same sensor node is selected as CH for the second time consecutively, then it may be a compromised node. Malicious node sends strong signal indicating it as the CH. This type of compromised node is identified by calculating the signal strength based on the distance. When a sensor node sends the join request to the CH and if it does not receive the TDMA (Time Division Multiple Access) schedule within certain period of time, then the CH may be a malicious node.

IDS can be categorized based on their detection method as Misuse and Anomaly. Based on the architecture, it can be classified as Host IDS (HIDS), Network IDS (NIDS), and Distributed IDS (DIDS). Based on the response, the IDS can be classified as active and passive, and in view of the decision making it can be classified as cooperative and autonomous. The authors list three works related to IDS on the LEACH protocol. Firstly, watchdog based IDS is proposed to capture the attack on each phase by applying the rules and secondly specification based IDS is proposed. The third method, namely, CUSUM IDS, is proposed based on the path construction, which uses normal path and malicious path information for detecting the intrusions [19]. In [20], the authors propose TESLA (Timed Efficient Stream Loss-Tolerant Authentication) based security mechanism to protect the LEACH protocol. The BS acts as a Key Distribution Center (KDC) and transfer the TESLA key to the sensor nodes periodically. The sensor nodes send the data along with the node membership certificate to the CH. By then, the CH aggregates and transfer the sensed data to the BS. The authors in [21] propose IDS by analyzing the detection rules in different architectures. They verified their work by identifying the sinkhole nodes on the Minroute protocol with less resource consumption. In [22], cryptographic based IDS are proposed to detect the sinkhole nodes. The BS verifies the digest value obtained from trustable forward path and from the trustable node to the destination. If the values are different, then the BS alerts the sensor nodes about the presence of sinkhole node. The authors in [23] propose centralized monitoring approach to detect the sinkhole node in the WSN. The monitoring node (or leader node) is randomly selected from the group

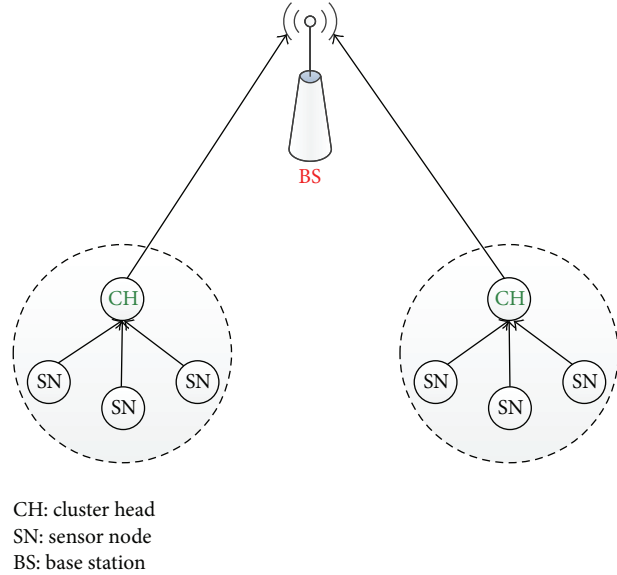


FIGURE 1: LEACH principle.

of sensor nodes. The leader node compares the node ID and location of the route nodes; if the node ID exists in the information table then it allows the transmission or alerts the other nodes about the intrusion. In [24], the authors propose a dynamic random password based IDS to detect the malicious activities in the WSN. The BS assigns node ID for each node and a password is generated dynamically for successful transmission of the data. A threshold based hierarchical IDS (THIDS) is proposed in [25] to detect the selective forwarding, black hole, and sinkhole attacks. Each sensor node has a local list called Isolate list to store the adversary's identities. When the sensor node does not receive any message from the CH for a period of time, then it is added to the Isolate list and sends a local alert message to the neighboring nodes about the presence of sinkhole node.

3. Research Background

3.1. Low Energy Adaptive Clustering Hierarchy (LEACH) Protocol. Hierarchical clustering is a method of arranging the nodes into a hierarchy of groups based on a weight function. Several hierarchical protocols exist in the literature and security is one of the prime concerns in these types of protocols [26]. LEACH is an energy efficient protocol that works based on hierarchical clustering. This protocol was proposed in 2002 by Heinzelman et al. [27]. They devised a new way of hierarchical clustering in WSNs that uses randomized rotation of CHs on the basis of sensor node (SN) properties such as energy and bandwidth. Unlike the usual clustering protocols where the CH continues to be the same node throughout the routing process, LEACH involves rotation of CHs in a dynamic and random manner so that energy is uniformly distributed to all the sensor nodes. Figure 1 shows the working principle of LEACH protocol.

The protocol works in two phases, namely, the setup phase and the steady state phase. In the first phase,

all the members of the sensor network participate in the election process by choosing a random number between 0 and 1 and if this number is less than the threshold value, $T(n)$, then that particular sensor node is elected as a CH [28].

$T(n)$ is calculated by the following equation:

$$T(n) = \frac{P}{1 - p(r \bmod (1/p))} \quad \text{if } n \in G \text{ else } T(n) = 0, \quad (1)$$

where “ p ” is the probability or desired percentage to become a CH, “ r ” is the current round, and “ G ” is the node set that has not been selected as a CH in the past $1/p$ rounds [28]. When the election is over, the entire set of elected CHs announces itself to be the CH by broadcasting an ADV (advertisement) packet. The remaining nodes find the CH which is nearer to them based on the minimum communication energy required. When the cluster is decided, the nodes send a JOIN-REQ (join request) packet to the corresponding CHs. At the end of setup phase, clusters are created and each CH allots time slot for its cluster nodes using TDMA. The fixing of time slots based on TDMA ensures that each node communicates with the CH during the allotted time slot to minimize collision. The steady state phase involves each node sending its data during its allotted time slot to the corresponding CH. The CH performs data aggregation and transmits it either to the BS or to its nearest CH when the BS is outside its transmission range. Thus CHs are the only nodes that interact directly with the BS. Only during their time slot, a cluster node has its radio in the active state; otherwise it gets into the sleep mode and thus sensor nodes reserve energy.

In addition to this, dynamic allocation of CHs ensures that every node has uniform energy and this makes LEACH as an energy efficient protocol. In LEACH, the sensors form local groups and a local CH is chosen randomly to serve as the local BS [27]. In this way, the CH position is rotated to assign CH to the cluster with the highest energy at any time. The energy load is balanced evenly so that none of the sensor nodes has drained off its energy fully. LEACH protocol is explained along with its advantages and disadvantages and the protocol performance is verified through simulation by considering the percentage of non-alive nodes [28].

3.2. Sinkhole Attack. Sinkhole attack is an active type of attack which focuses on the routing pattern of a protocol. The compromised node (CN) acts as a sinkhole and attracts all the traffic towards itself [10]. The compromised node grabs attention from the other nodes by establishing itself to have a high value with respect to the routing metric [11]. As a result, the intruder gets control over the packets and proceeds to launch further attacks like black hole, selective forwarding, altering packets, and so on. Various types of attacks based on the location are also present in the literature and existing schemes like multipath routing, hashing, cryptography, key distribution, localization, and IDS are used to counter these types of routing attacks [29]. Various methods are present to launch the sinkhole attack, either by directly giving false information about the routing metric to the sender nodes or by using wormhole attack. The wormhole threat creates a separate link from the normal network link and starts

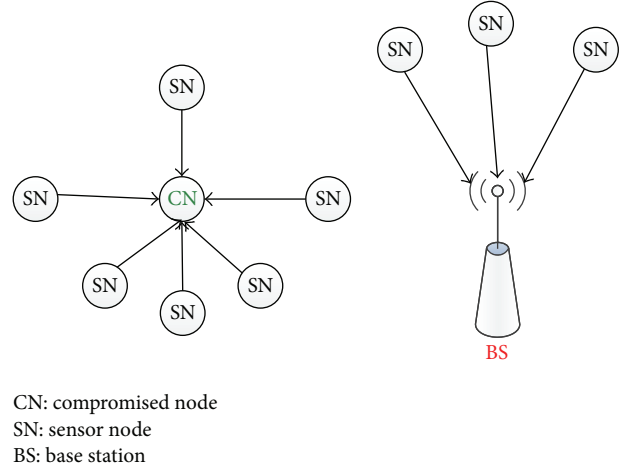


FIGURE 2: Sinkhole attack.

forwarding the data between them. Either of the nodes in the wormhole link can be made as a sinkhole node and further attacks can be launched. Figure 2 shows sinkhole attack (launched by compromised node).

The compromised node sends the fake routing information to the normal sensor nodes to transmit their sensed data. The compromised node can drop the packets completely and this process of threat is called black hole attack [30]. The sinkhole node can also be used as a platform for launching other threats like forwarding the packets selectively or deleting some fields in the packet. This kind of attack is called selective forwarding. This research work focuses on the analysis of adverse effects of the sinkhole attack on LEACH protocol and develops an efficient defense mechanism to reduce the adverse effects of the sinkhole attack to the network. The sinkhole attack can be launched on various routing protocols by falsifying the routing metric. The sinkhole attack in [10] is launched in the Minroute protocol by giving false information about the link quality which is used as a routing metric by the protocol. The compromised node gives the high link quality to make other nodes forward their data to it.

The sinkhole attack is detected by identifying the intrusion region using geostatistical hazard model and distributed monitoring approach. This method is computationally expensive [13]. In case of LEACH protocol, the sinkhole attack can be launched using the CHs. The compromised node projects itself with high energy value and makes it to be selected as the CH. This compromised CH acts as a sinkhole node and performs the attack by dropping or altering the sensed data which is received from its cluster members. The sinkhole attack can be launched in other routing protocols and an efficient defense mechanism is needed to counter this attack.

3.3. Research Motivation. Table 1 classifies the attacks based on the layers of the WSN protocol stack, where the network layer has many potential vulnerabilities like sinkhole attack, black hole attack, selective forwarding, Sybil, HELLO flood, wormhole, and so on. The purpose of routing attack is to

create a serious threat to the sensor network. The sinkhole is one of the most susceptible threats to the sensor network as referred to in [9, 11, 21, 23, 25–27]. It can be extended further with the attacks like selective forwarding, black hole, and HELLO flood to devastate the network transmission [31–33]. Thus, this paper places its major concern on the sinkhole nodes, since it is more vulnerable than the other security threats. In addition, it is interesting to study the effects of the attack to develop the defense mechanism.

As LEACH is a hierarchical protocol, CH plays the major role for data transmission, where the CH is compromised as a sinkhole node to disrupt the condition of the network [34]. The protocol, like LEACH, has had several extensions like LEACH-C [35], LEACH-F [36], LEACH-B [37], LEACH-E [38], LEACH-M [39], MH-LEACH [40], I-LEACH [41], V-LEACH [42], and so on. These extended versions focus on minimizing the energy consumption and reducing the transmission overhead [43]. In addition, the extended versions of the LEACH protocol have not had its concern on the feature like intrusion detection. The existing security mechanisms of the LEACH protocol are generally classified into cryptographic based methods and non-cryptographic methods. The cryptographic methods are S-LEACH [44], Armor-LEACH [45], R-LEACH [46], MS-LEACH [47], and Sec-LEACH [48]. The non-cryptographic methods are signal strength based approach [49] and TM-LEACH [50]. In S-LEACH, the detection of sinkhole and selective forwarding attacks were dealt. Since it is a cryptographic method, it increases the computational overhead of the networks. Recently, the S-LEACH has been extended as MS-LEACH, though MS-LEACH still lacks in throughput efficiency and energy consumption. The non-cryptographic method is based on signal strength and trust-value security. But, it does not deal with the routing attacks in WSN. Above all, the effects of the sinkhole attack on the LEACH protocol is still not present on lightweight non-cryptographic method. Thus, this research focuses on the development of non-cryptographic IDS in turn to reduce the adverse effects with minimum resource utilization.

4. System Design

Security is the prime concern in a wireless network and the sinkhole attacks are so vicious that they overcome all the other attacks. The effort of providing security was channelized in studying the possibility of sinkhole attack in a sensor network having LEACH as the routing protocol, the attack effects, and developing an IDS to minimize the adverse effects. Important Notations Used Section shows the notations used and their description.

4.1. Problem Description. In LEACH protocol, the energy value is the deciding factor that selects the CH and it has a threshold dependency. The initial study is to understand the clustering pattern of the network. Thereafter, the intruder launches the attack, like sinkhole attack in the LEACH routing based network, with the help of a cluster of nodes. The parameters, namely, number of clusters (nc) and cluster members (SN_i), are observed to predict the values. During

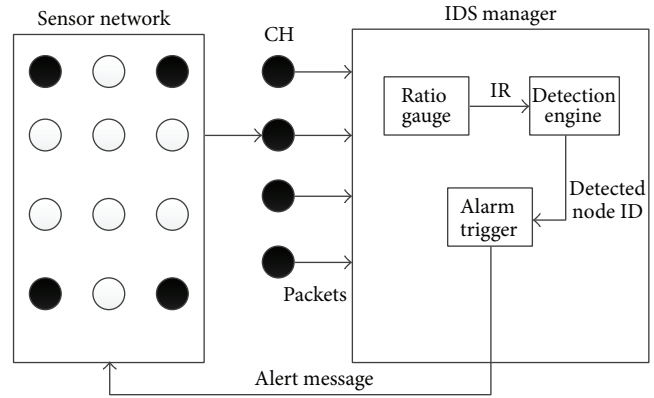


FIGURE 3: IDS architecture.

the attack operation, the values are considered to compromise the CHs in different locations. Since the routing attacks emerge as a problematic issue for the WSN, the intruders launch the attack in two different ways.

4.1.1. Launching of Sinkhole Attack. The launching of attacks is classified in two different ways. The former launching is to use a set of CH nodes to launch a coordinated sinkhole attack. The objective of this attack is to compromise “ nc ” nodes that are available across the network, such that each compromised node belongs to the cluster. In the coordinated sinkhole attack, the number of clusters is equal to the number of compromised nodes. So, the compromised nodes gain the control of normal sensor nodes to project its energy values as above the threshold to become a CH. In the steady state phase, the normal nodes transmit their data packets to the compromised CHs and thus every compromised CH or sinkhole node can drop or manipulate the packets to complete the security breaching. The latter launching is to introduce a compromised sinkhole attack to manipulate the data of its cluster members in the network. In order to launch the sinkhole attack, CH is compromised for each round of selection process instead of compromising “ nc ” cluster. In such scenario, one CH would be malicious to act as a sinkhole. Though it has limited performance loss for each data transmission, the malicious activity is extended further in the compromised sinkhole attack. Hence, it has been addressed as a challenging problem for the sensor networks. However, such attacks can be detected effectively by the mechanism of IDS.

4.2. IDS Architecture. Figure 3 gives an overview of working principle of the proposed IDS. Recent routing protocols face security issues in the presence of multiple sink or BS and node mobility [51]. The proposed IDS works fine in the presence of multiple sinks by placing the detection agent on each sink. The proposed work assumes that the compromised nodes, that is, the sinkhole nodes, blindly drop or selectively forward the packets which are received from the normal sensor nodes. The CH collects the data from the cluster member and it is later analyzed by the BS. The IDS agent that runs in the BS

```

Begin
 $S_n$  is the sensor network and  $PT_i$  be the total packets transmitted by the  $i$ th CH in  $S_n$ 
 $PR_i$  is the total packets received by the  $i$ th CH in  $S_n$ 
 $N_i$  is the Cluster head Node ID
 $P_i$  is the Intrusion ratio for the  $i$ th CH
Repeat
  Time_delay (100)
  For  $\forall (C_i)$ 
    Receive ( $PR_i$ ,  $PT_i$  and  $N_i$ ) packets from the CHs'
    Calculate  $P_i$  where  $P_i = PR_i/PT_i$ 
    If  $P_i$  tends to  $\infty$  then
      Corresponding  $N_i$  is the sinkhole node
      Isolate  $N_i$ 
      Send warning message to the remaining cluster member nodes about  $N_i$ 
    Else
      Corresponding  $N_i$  is the normal CH
    End if
  End for
Until the nodes transmission process completes
End

```

ALGORITHM 1: IDS Algorithm for Sinkhole Attack.

receives the packets by overhearing the transmission of the cluster members and CH nodes. The IDS agent contains ratio gauge module which calculates the intrusion ratio (IR) from the values obtained from the network. The packet received (PR_i), packet transmitted (PT_i), and CH node ID's (N_i) values are used to calculate the IR. The ratio gauge sends the IR value to the detection engine. The detection engine triggers the alarm which depends upon the IR value indicating the presence of compromised node.

The algorithm for the intrusion detection process is given in Algorithm 1.

4.2.1. Algorithm Description. The IDS agent module runs in the BS to identify the intrusion by analyzing the data packets that consists of PR_i , PT_i , and N_i periodically. The packet transmission value of CH (PT_i), the packet reception value of CH (PR_i), and the CH node identification of the CH (N_i) are used to validate the intrusion ratio (IR) as numeric or not by the IDS agent. If the ratio of PR_i to PT_i is numeric, it means that the packet is not completely dropped to ensure "the malicious activity is not existing." Otherwise (IR is infinity), the corresponding CH is a sinkhole node which had dropped the data packets completely that would lead to black hole attack. On the other hand, if there is a huge difference between PR_i and PT_i values, it infers that there may be a possibility of selective forwarding attack.

The purpose of the above strategy is to minimize the intrusion ratio so that the intruder node can be isolated in the next round of data transmission and blocked from the CH selection process by the BS. The proposed IDS mechanism alerts the respective cluster members regarding the presence of sinkhole node to stop the further data transmission. Moreover, this algorithm has much less computation to detect the sinkhole node from the available information (local) and

it also increases the energy efficiency of the network by the quick identification of the compromised nodes. Since the proposed IDS mechanism has less communication overhead between the sensor networks and the BS, the ratio gauge calculation is simple to make the computation easier which reduces the computational complexity to the further extent. Even though the node density of the sensor network is increased, the proposed IDS mechanism works efficiently to alert the threat deduction. The proposed IDS has much less storage since its values are removed from the buffer after computing the IR value.

4.3. Intrusion Detection Model. A simple mathematical model is constructed to verify the effectiveness of the IDS algorithm. The sensor network (S_n) consists of many sensor nodes uniformly placed over the network. Let S_n consist of SN_1, SN_2, \dots , and SN_n sensor nodes in the network. Let C_i be the cluster which contains the sensor nodes as its members. The N_i is the CH of a particular cluster C_i . The CHs are selected in the setup phase and the sensor nodes join the cluster depending upon the transmission range. The sensor nodes SN_i start sensing the physical phenomenon like temperature, humidity, movement tracking, and so on. The packets received (PR_i) and packets transmitted (PT_i) value of a particular CH (N_i) are calculated by the BS through overhearing the transmission of the cluster members and the CH.

Mathematically the intrusion ratio (IR) is represented by the following equation:

$$P_i = \frac{PR_i}{PT_i}. \quad (2)$$

The IR value (P_i) can take any of the following two values: $P_i = \infty$ or $P_i = n$. From the value of P_i , the IDS agent

decides whether a CH is malicious or normal. If the IR value is a numeric value (n), then it denotes that the packets are not fully dropped. If the IR value is an infinite value (∞), it denotes that $PT_i = 0$ which indicate the presence of some malicious activity. The following equation presents the values of the IR(P_i):

$$P_i = \begin{cases} n \rightarrow N_i \text{ is a normal node, } \forall n \in \text{integer} \\ \infty \rightarrow N_i \text{ is sinkhole node.} \end{cases} \quad (3)$$

Consider an example sensor network with four clusters. The sensor nodes $SN_1, SN_2, SN_3,$ and SN_4 form a cluster C_1 and similarly clusters $C_2, C_3,$ and C_4 are constructed. The cluster members transmit their sensed data to their corresponding CHs (N_i). The data is temporarily stored in the buffer of N_i . For the CH (N_1) in the cluster C_1 , the IR is calculated as follows:

$$P_1 = \frac{PR_1}{PT_1}. \quad (4)$$

For the Other Clusters. The IR values can be calculated as $P_2 = PR_2/PT_2, P_3 = PR_3/PT_3$ and $P_4 = PR_4/PT_4$. To explain the model, the IR calculation is classified into two different cases with sample values to show the possibility, such as under attack/no attack.

Case 1 (under attack). Consider 4 cluster members $SN_1, SN_2, SN_3,$ and SN_4 and a CH N_1 , given some sample input values, the IR is calculated. Let the PR_1 value of the CH $N_1 = 80$ and the PT_1 value of CH $N_1 = 0$. The IR value of the CH N_1 is calculated from (4) as $P_1 = 80/0 = \infty$ which indicate that the CH N_1 is a sinkhole node, since it drops all the 80 packets ($PT_1 = 0$).

Case 2 (no attack). Consider 4 cluster members $SN_1, SN_2, SN_3,$ and SN_4 and a CH N_1 . Given some sample input values, the IR is calculated. Let the PR_1 value of the CH $N_1 = 80$ and the PT_1 value of CH $N_1 = 76$. The IR value of the CH N_1 is calculated from (4) as $P_1 = 80/76 = 1.05$, which is a numeric value and indicates that the CH N_1 is a normal node since it forwards the packets. Similarly, the process of IR estimation is done to all the CHs in the network. Minor packet loss occurs due to network link conditions. The IR value is calculated from the local data collected by the ratio gauge present in the IDS agent and hence the proposed IDS reduce the energy consumption to acquire the global data access.

5. Simulation and Analysis

5.1. Simulation Process. The proposed work was simulated using TETCOS NETSIM. Table 2 shows the simulation setup. The following are the assumptions for simulation of the proposed IDS.

- (1) BS has the highest energy resource.
- (2) All the sensor nodes are static.

TABLE 2: Simulation setup.

Sensor nodes	36
Agent nodes	1
Base station	1
Transmission range	100 m
Transmission power	100 mW
Frame retries	0
Arrangement	Uniform

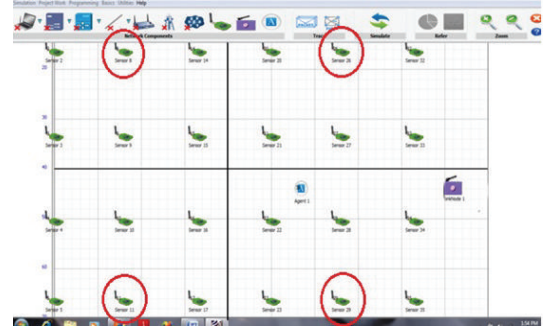


FIGURE 4: Initial scenario.

- (3) All the sensor nodes transfer data in the allocated frame.
- (4) Compromised nodes have higher energy level than normal sensor nodes.

The initial setup consists of 4 clusters of 9 sensors inclusive of the one being the CH of each cluster and the members are increased for the simulation process. The scenario highlighting the initial set of CHs, namely, sensors 8, 11, 26, and 29, is shown in Figure 4.

The simulation of LEACH in NETSIM resulted in the packet transfers depicted in Figures 5, 6, and 7. Figure 5 shows the normal packet transfer operation from the sensor node to CH 8. In Figure 6, the transmission happens between CHs 8 and 26 because the BS is not within the transmission range of CH 8. Figure 7 shows the packet transmission of CH 26 to the final destination BS.

When a coordinated sinkhole attack is launched in the above scenario, all the CHs drop the packets and no packets are received by the BS. However, sinkhole attack on a single CH showed that the BS received packets from all the CHs except the compromised CH. The following section shows the results of simulation depicting the number of packets transmitted and received. The simulation results are graphically represented in which the x -axis denotes the CH and y -axis the packets. Figure 8 shows the values recorded during the normal LEACH operation.

In Figure 9, the values recorded during cooperative sinkhole attack are shown, where the BS receives no data. Figure 10 shows the values recorded when a single CH, that is, sensor node 8, is attacked. Note that the number of packets transmitted by sensor node 8 is 0.

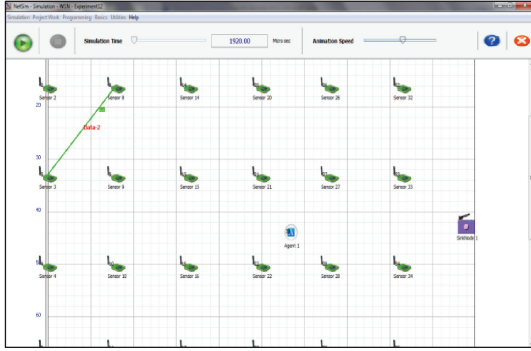


FIGURE 5: Node sending data to CH 8.

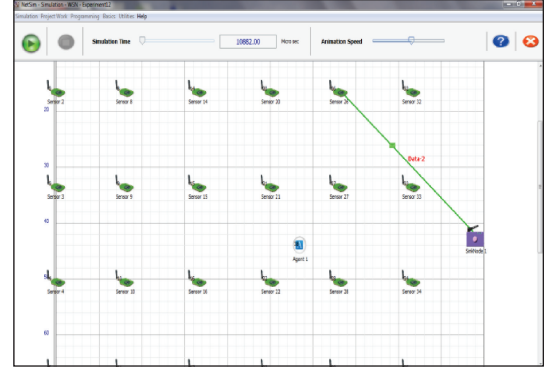


FIGURE 7: CH 26 sending data to BS.

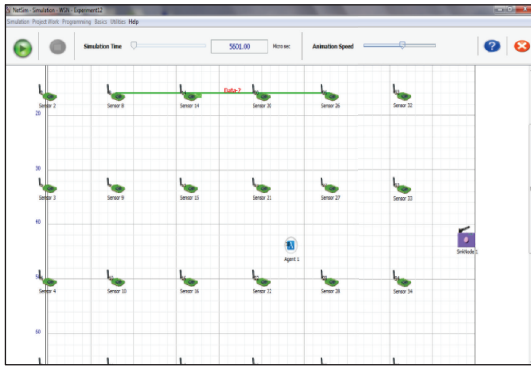
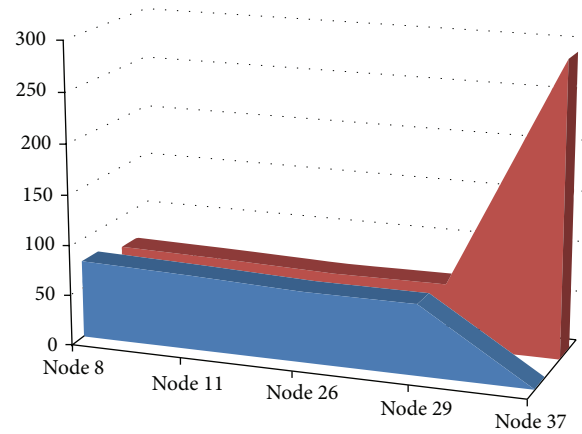


FIGURE 6: CH 8 sending data to BS.



■ Packets successfully transmitted
 ■ Packets received from cluster nodes

FIGURE 8: Normal LEACH operation.

5.2. *Analysis.* To analyze the proposed algorithm, the recorded values during the simulation are taken. Table 3 lists the values of the LEACH normal operation and also the values recorded after launching the coordinated sinkhole attack and attack on a single CH, respectively.

The nodes in Table 3, N_8 , N_{11} , N_{26} , and N_{29} , denote the CH nodes and the node N_{37} indicates the BS which does not forward any data, so its PT_i value is 0. For the analysis of the results, three different scenarios are considered.

5.2.1. *Results and Discussion.* The results are analyzed in three different scenarios, that is, normal operation, coordinated sinkhole attack, and single node sinkhole attack.

LEACH Normal Operation. Consider CH 8: applying the algorithm for the values listed in Table 3, the IR is calculated using (2):

$$P_8 = \frac{67}{77}, \quad (5)$$

$P_8 = 0.87$, where the ratio value is a numeric value which shows sensor 8 is a normal node, since the packets are transmitted.

Consider CH 11: the IR value is calculated using (2):

$$P_{11} = \frac{65}{74}, \quad (6)$$

$P_{11} = 0.88$, which indicate a numeric value which proves node N_{11} is a normal node. This process is repeated for other

TABLE 3: Comparison of normal and threat operation in LEACH protocol.

Cluster head (CH)	N_8		N_{11}		N_{26}		N_{29}		N_{37} (BS)	
	PT_i	PR_i	PT_i	PR_i	PT_i	PR_i	PT_i	PR_i	PT_i	PR_i
LEACH normal operation	77	67	74	65	70	62	70	63	0	291
Coordinated sinkhole attack	0	67	0	65	0	59	0	62	0	0
Sinkhole attack on CH 8	0	67	74	65	69	61	69	60	0	212

two nodes N_{29} and N_{26} . The result is a numeric value which proves them as attack-free nodes.

Coordinated Sinkhole Attack. Calculate the IR for CH 8 from Table 3 using (2):

$$P_8 = \frac{67}{0}, \quad (7)$$

$$P_8 = \infty.$$

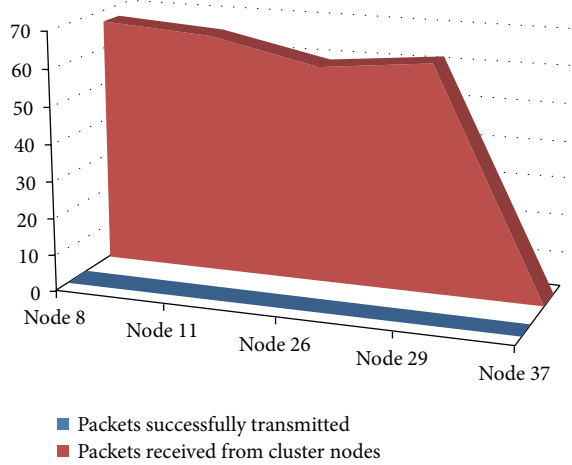


FIGURE 9: Coordinated sinkhole attack.

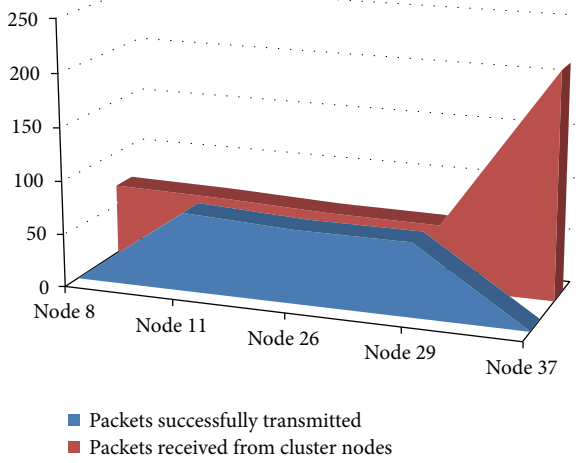


FIGURE 10: Sinkhole attack on CH 8.

The IR value is “ ∞ ” which shows packet dropping had occurred ($PT_i = 0$) and hence CH 8 is compromised.

Consider the CH 11: from (2),

$$P_{11} = \frac{65}{0}, \quad (8)$$

$P_{11} = \infty$, which shows the occurrence of malicious activity. This process is repeated for the other two nodes N_{29} and N_{26} . The result is an infinite value (∞) which proves them as sinkhole nodes. Hence coordinated sinkhole attack is identified.

Sinkhole Attack on CH 8. Calculating the IR for values in Table 3 for CH 8, from (2)

$$P_8 = \frac{67}{0}, \quad (9)$$

$P_8 = \infty$, which shows the presence of a sinkhole node.

Consider CH 11. The IR value is calculated using (2),

$$P_{11} = \frac{65}{74}, \quad (10)$$

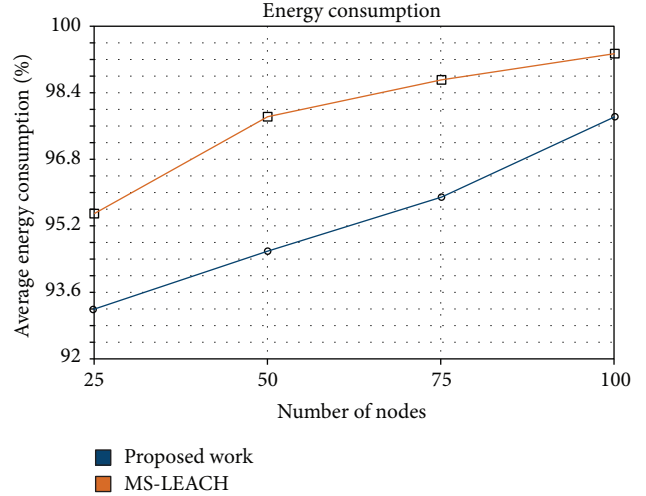


FIGURE 11: Average energy consumption rate comparison.

$P_{11} = 0.87$, which indicate an integer proving node N_{11} is a normal node. This process is repeated for other two nodes N_{29}, N_{26} . The result is an integer which proves them as attack-free nodes and these analyses prove that the attack is launched only on CH 8. The above analysis proves the correctness of the proposed algorithm.

5.3. Performance Evaluation. The recent method to detect the sinkhole attack in LEACH based network is MS-LEACH which is an extension of S-LEACH. The S-LEACH is the first security extension of LEACH and it follows the SPIN building blocks for its security features. MS-LEACH adapts pairwise key establishment method to provide data confidentiality and authentication for cluster member to CH. This protocol outperforms the S-LEACH in terms of power consumption, network throughput, and lifetime [47]. To compare our proposed scheme with the recent existing work, MS-LEACH is chosen. Since this protocol is the extension of S-LEACH and also provides detection of sinkhole attack, it would be a better work for our comparative study. For the performance comparison, three main metrics are used, namely, average energy consumption, average network lifetime, and average network throughput. The following section gives a brief comparison of the performance of the proposed scheme with the existing work.

5.3.1. Average Energy Consumption. The ratio of the energy consumed by all the sensor nodes to the amount of the total startup energy is the average energy consumed by the nodes. Figure 11 shows that the proposed scheme consumes around 2% less energy compared to MS-LEACH.

The proposed scheme allows the computation to be performed on the BS. Since the BS is the powerful energy resource, it performs all the computation effectively. In the proposed work, the cluster members and CH are not involved in the computation process, so the energy consumption by the sensor nodes is very minimal and proves to be energy efficient method.

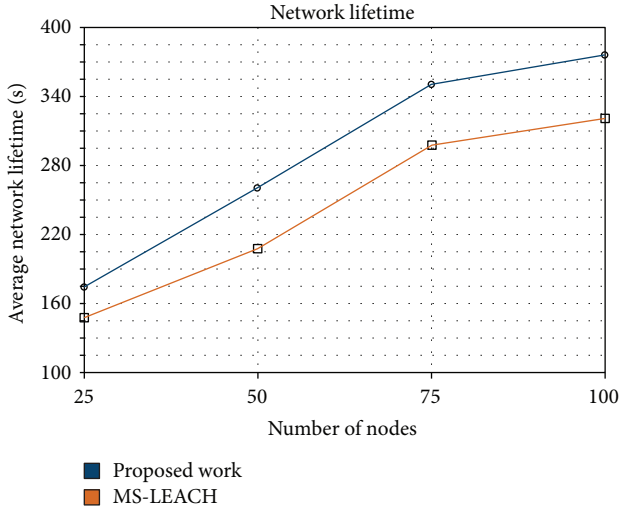


FIGURE 12: Average network lifetime comparison.

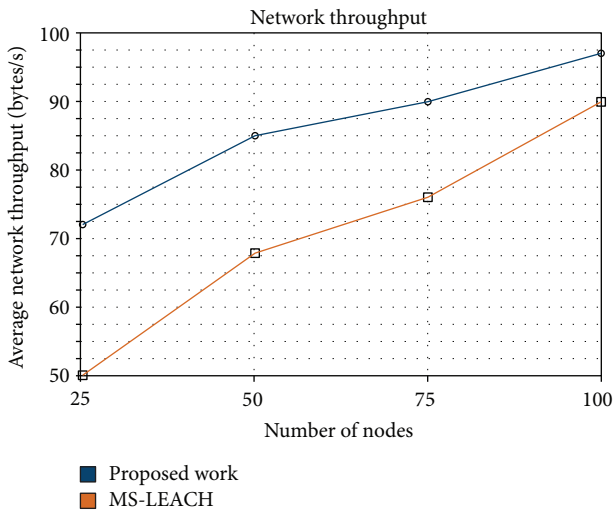


FIGURE 13: Average network throughput comparison.

5.3.2. *Average Network Lifetime.* The average network lifetime is the total time period between the start of the simulation process and the termination of the process due to energy depletion. Figure 12 shows the comparison of network lifetime between the proposed scheme and the MS-LEACH protocol.

The proposed scheme holds network lifetime of about 52% more than MS-LEACH which makes the network extend the lifetime and makes the sensor nodes alive for a long period.

5.3.3. *Average Network Throughput.* The network throughput is the ratio of the total data received to the certain period of time. The proposed scheme detects the sinkhole nodes at the earliest and minimizes the packet drop rate. So, the network throughput increases gradually compared to MS-LEACH.

Figure 13 shows that the proposed scheme increases the network throughput by 15% more than MS-LEACH since

it uses lightweight IDS to detect the intrusion quickly. The throughput is the important metric to compare the effectiveness of the proposed work with the existing work, since the proposed IDS deals with the packet dropping attack.

In summary, the proposed work outperforms the existing MS-LEACH in terms of less energy consumption, extended network lifetime, and increased network throughput.

6. Conclusion and Future Work

WSNs are easily prone to security breach like sinkhole attack. Thus an IDS mechanism has been proposed that identifies such attacks on LEACH protocol and alerts the normal sensor node to reduce the data loss rate. The TETCOS NETSIM simulator has been used for the analysis, where the sinkhole attack and IDS were launched. The simulation result shows that the vulnerability like sinkhole attacks on LEACH drops all the transmitted packets across the CH. The proposed IDS captured the sinkhole nodes with the minimum computation and alerted the normal sensor nodes. Since the computation of proposed IDS is simple, it consumes less energy, whereas the network lifetime can be extended as compared to the existing work, namely, MS-LEACH. In addition, the numerical analysis proves that the proposed IDS can achieve minimum computational overhead and less energy consumption. In future, the proposed algorithm can be extended towards the detection of selective forwarding attack which alters fragment of the data and snooze attack, respectively.

Important Notations Used

- nc: Number of clusters
- p : Probability of sensor node to be elected as CH
- P_i : Intrusion ratio (IR)
- PR_i : Packets received by the i th CH
- PT_i : Packets transmitted by the i th CH
- N_i : CH node ID's
- SN_i : Sensor node or cluster members
- S_n : Sensor network
- C_i : Cluster.

Conflict of Interests

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

Acknowledgment

The authors would like to acknowledge SASTRA University for the great support and assistance rendered to carry out this research work.

References

- [1] E. C. H. Ngai, J. Liu, and M. R. Lyu, "An efficient intruder detection algorithm against sinkhole attacks in wireless sensor

- networks,” *Computer Communications*, vol. 30, no. 11-12, pp. 2353–2364, 2007.
- [2] 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.
 - [3] G. Padmavathi and D. Shanmugapriya, “A survey of attacks, security mechanisms and challenges in wireless sensor networks,” *International Journal of Computer Science and Information Security*, vol. 4, no. 1-2, 2009.
 - [4] N. A. Alrajeh, S. Khan, and B. Shams, “Intrusion detection systems in wireless sensor networks: a review,” *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 167575, 7 pages, 2013.
 - [5] J. A. Chaudhry, U. Tariq, M. A. Amin, and R. G. Rittenhouse, “Dealing with sinkhole attacks in wireless sensor networks,” *Advanced Science and Technology Letters*, vol. 29, pp. 7–12, 2013.
 - [6] M. Dener, “Security analysis in wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 303501, 9 pages, 2014.
 - [7] T. Singh and H. Kaur Arora, “Detection and correction of sinkhole attack with novel method in WSN using NS2 tool,” *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 2, pp. 32–35, 2013.
 - [8] S. A. Salehi, M. A. Razzaque, P. Naraei, and A. Farrokhtala, “Detection of sinkhole attack in wireless sensor networks,” in *Proceedings of the 3rd IEEE International Conference on Space Science and Communication (IconSpace '13)*, pp. 361–365, Melaka, Malaysia, July 2013.
 - [9] P. Maidamwar and N. Chavhan, “Impact of wormhole attack on performance of LEACH in wireless sensor networks,” *International Journal of Computer Networking, Wireless and Mobile Communications*, vol. 3, no. 3, pp. 21–32, 2013.
 - [10] I. Krontiris, T. Dimitriou, T. Giannetsos, and M. Mpasoukos, “Intrusion detection of sinkhole attacks in wireless sensor network,” in *Proceedings of the 3rd International Workshop on Algorithmic Aspects of Wireless Sensor Networks (AlgoSensors '07)*, vol. 4837, pp. 150–161, Wrocław, Poland, 2007.
 - [11] S. Iqbal, S. P. Aravind Srinivas, G. Sudharsan, and S. S. Kashyap, “Comparison of different attacks on LEACH protocol in WSN,” *International Journal of Electrical, Electronics and Data Communication*, vol. 8, no. 8, pp. 16–19, 2014.
 - [12] X. Deng, R. Wu, W. Wang, and R. Bu, “An intrusion detection system for cluster based wireless sensor networks,” *Information Technology Journal*, vol. 12, no. 9, pp. 1764–1771, 2013.
 - [13] H. Shafiei, A. Khonsari, H. Derakhshi, and P. Mousavi, “Detection and mitigation of sinkhole attacks in wireless sensor networks,” *Journal of Computer and System Sciences*, vol. 80, no. 3, pp. 644–653, 2014.
 - [14] V. K. Jatav, M. Tripathi, M. S. Gaur, and V. Laxmi, “Wireless sensor networks: attack models and detection,” in *Proceedings of IACSIT Hong Kong Conferences*, vol. 30, pp. 144–150, 2012.
 - [15] M. Bahekmat, M. H. Yaghmae, A. S. H. Yazdi, and S. Sadeghi, “A novel algorithm for detecting sinkhole attacks in WSNs,” *International Journal of Computer Theory and Engineering*, vol. 4, no. 3, pp. 418–421, 2012.
 - [16] E.-N. Huh, T. H. Hai, and M. Jo, “A lightweight intrusion detection framework for wireless sensor networks,” *Wireless Communications and Mobile Computing*, vol. 10, no. 4, pp. 559–572, 2010.
 - [17] M. R. Rohbanian, M. R. Kharazmi, A. Keshavarz- Haddad, and M. Keshtgary, “Watchdog-LEACH: a new method based on LEACH protocol to secure clustered wireless sensor networks,” *Advances in Computer Science*, vol. 2, no. 3, pp. 105–117, 2013.
 - [18] S. Lee, Y. Lee, and S.-G. Yoo, “A specification based intrusion detection mechanism for the LEACH protocol,” *Information Technology Journal*, vol. 11, no. 1, pp. 40–48, 2012.
 - [19] S. Gupta and V. Grover, “Survey of intrusion detection techniques in LEACH,” *International Journal of Computer Trends and Technology*, vol. 17, no. 4, pp. 166–171, 2014.
 - [20] S. Ramachandran and V. Shanmugam, “An approach to secure leach using tesla based certificate,” *Life Science Journal*, vol. 10, no. 2, pp. 1018–1027, 2013.
 - [21] M. A. Rassam, A. Zainal, M. A. Maarof, and M. Al-Shaboti, “A sinkhole attack detection scheme in minroute wireless sensor networks,” in *Proceedings of the 1st IEEE International Symposium on Telecommunication Technologies (ISTT '12)*, pp. 71–75, 2012.
 - [22] S. Sharmila and G. Umamaheswari, “Detection of sinkhole attack in wireless sensor networks using message digest algorithms,” in *Proceedings of the International Conference on Process Automation, Control and Computing (PACC '11)*, pp. 1–6, Coimbatore, India, July 2011.
 - [23] D. U. S. Rajkumar and R. Vayanaperumal, “A leader based monitoring approach for sinkhole attack in wireless sensor network,” *Journal of Computer Science*, vol. 9, no. 9, pp. 1106–1116, 2013.
 - [24] A. Thomas Paul Roy and K. Balasubadra, “DRPGAC: detecting and preventing malicious activities in wireless sensor networks,” *Journal of Theoretical and Applied Information Technology*, vol. 69, no. 1, pp. 143–150, 2014.
 - [25] S. Sahraoui and S. Bouam, “Secure routing optimization in hierarchical Cluster-Based wireless sensor networks,” *International Journal of Communication Networks and Information Security*, vol. 5, no. 3, pp. 178–185, 2013.
 - [26] S. Sharma and S. K. Jena, “A survey on secure hierarchical routing protocols in wireless sensor networks,” in *Proceedings of the International Conference on Communication, Computing and Security (ICCCS '11)*, pp. 146–151, February 2011.
 - [27] 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.
 - [28] A. Kaur and S. Saini, “Simulation of low energy adaptive clustering hierarchy protocol for wireless sensor network,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 7, pp. 1316–1320, 2013.
 - [29] M. Elhoseny, H. K. El-Minir, A. M. Riad, and X. Yuan, “Recent advances of secure clustering protocols in wireless sensor networks,” *International journal of Computer Networks and Communications Security*, vol. 2, no. 11, pp. 400–413, 2014.
 - [30] S. Athmani, D. E. Boubiche, and A. Bilami, “Hierarchical energy efficient intrusion detection system for black hole attacks in WSNs,” in *Proceedings of the World Congress on Computer and Information Technology*, pp. 1–5, June 2013.
 - [31] M. Tripathi, M. S. Gaur, and V. Laxmi, “Comparing the impact of black hole and gray hole attack on LEACH in WSN,” in *Proceedings of 4th International Conference on Ambient Systems, Networks and Technologies (ANT '13)*, vol. 19, pp. 1101–1107, June 2013.
 - [32] A. Jangra and P. Swati, “Securing LEACH protocol from sybil attack using jakes channel scheme (JCS),” in *Proceeding of International Conference on Advances in ICT for Emerging Regions*, pp. 79–87, Colombo, Sri Lanka, September 2011.

- [33] S. Magotra and K. Kumar, "Detection of HELLO flood attack on LEACH protocol," in *Proceedings of the 4th IEEE International Advance Computing Conference (IACC '14)*, pp. 193–198, February 2014.
- [34] M. Sankar, M. Sridar, and M. Rajani, "Performance evaluation of LEACH protocol in wireless network," *International Journal of Scientific & Engineering Research*, vol. 3, no. 1, 2012.
- [35] W. Xinhua and W. Sheng, "Performance comparison of LEACH and LEACH-C protocols by NS2," in *Proceedings of the 9th International Symposium on Distributed Computing and Applications to Business Engineering and Science (DCABES '10)*, pp. 254–258, Hong Kong, China, August 2010.
- [36] W. Heinzelman, *Application-specific protocol architectures for wireless networks [Ph.D. dissertation]*, Massachusetts Institute of Technology, 2000.
- [37] M. Tong and M. Tang, "LEACH-B: an improved LEACH protocol for wireless sensor network," in *Proceedings of 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM '10)*, pp. 1–4, Chengdu, China, September 2010.
- [38] L. Qing, Q. Zhu, and M. Wang, "Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks," *Computer Communications*, vol. 29, no. 12, pp. 2230–2237, 2006.
- [39] D.-S. Kim and Y.-J. Chung, "Self-organization routing protocol supporting mobile nodes for wireless sensor network," in *Proceedings of the 1st International Multi-Symposiums on Computer and Computational Sciences (IMSCCS '06)*, vol. 2, pp. 622–626, Hanzhou, China, June 2006.
- [40] R. V. Biradar, D. S. R. Sawant, D. R. R. Mudholkar, and D. V. C. Patil, "Multi-Hop routing in self-organizing wireless sensor networks," *International Journal of Computer Science Issues*, vol. 8, no. 1, pp. 154–164, 2011.
- [41] N. Kumar and J. Kaur, "Improved LEACH protocol for wireless sensor networks," in *Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '11)*, pp. 1–5, September 2011.
- [42] N. Sindhvani and R. Vaid, "V LEACH: an energy efficient communication protocol for WSN," *Mechanica Confab*, vol. 2, no. 2, pp. 79–84, 2013.
- [43] H. Dhawan and S. Waraich, "A comparative study on LEACH routing protocol and its variants in wireless sensor networks: a survey," *International Journal of Computer Applications*, vol. 95, no. 8, pp. 21–27, 2014.
- [44] A. C. Ferreira, M. A. Vilaça, L. B. Oliveira, E. Habib, H. C. Wong, and A. A. Loureiro, "On the security of cluster-based communication protocols for wireless sensor networks," in *Proceedings of the 4th IEEE International Conference on Networking (ICN '05)*, vol. 3420 of *Lecture Notes in Computer Science*, pp. 449–458, 2005.
- [45] M. A. Abuhelaleh, T. M. Mismar, and A. A. Abuzneid, "Armor-LEACH—energy efficient, secure wireless networks communication," in *Proceedings of the 17th International Conference on Computer Communications and Networks (ICCCN '08)*, pp. 1–7, St. Thomas, Virgin Islands, USA, August 2008.
- [46] K. Zhang, C. Wang, and C. Wang, "A secure routing protocol for cluster-based wireless sensor networks using group key management," in *Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '08)*, pp. 1–5, October 2008.
- [47] M. El-Saadawy and E. Shaaban, "Enhancing S-LEACH security for wireless sensor networks," in *Proceedings of the IEEE International Conference on Electro/Information Technology (EIT '12)*, pp. 1–6, IEEE, Indianapolis, Ind, USA, May 2012.
- [48] L. B. Oliveira, A. Ferreira, M. A. Vilaça et al., "SecLEACH-on the security of clustered sensor networks," *Signal Processing*, vol. 87, no. 12, pp. 2882–2895, 2007.
- [49] V. Pal, S. Aishwarya, and S. Jain, "Signal strength based HELLO flood attack detection and prevention in wireless sensor networks," *International Journal of Computer Applications*, vol. 62, no. 15, pp. 1–6, 2013.
- [50] W. Wang, F. Du, and Q. Xu, "An improvement of LEACH routing protocol based on trust for wireless sensor networks," in *Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '09)*, pp. 1–4, September 2009.
- [51] A. M. El-Semary and M. M. Abdel-Azim, "New trends in secure routing protocols for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 802526, 16 pages, 2013.

Research Article

A Data Processing Middleware Based on SOA for the Internet of Things

Feng Wang, Liang Hu, Jin Zhou, and Kuo Zhao

College of Computer Science and Technology, Jilin University, Changchun 130012, China

Correspondence should be addressed to Kuo Zhao; zhaokuo@jlu.edu.cn

Received 17 September 2014; Revised 11 January 2015; Accepted 27 January 2015

Academic Editor: Fei Yu

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

The Internet of Things (IoT) emphasizes on connecting every object around us by leveraging a variety of wireless communication technologies. Heterogeneous data fusion is widely considered to be a promising and urgent challenge in the data processing of the IoT. In this study, we first discuss the development of the concept of the IoT and give a detailed description of the architecture of the IoT. And then we design a middleware platform based on service-oriented architecture (SOA) for integration of multisource heterogeneous information. New research angle regarding flexible heterogeneous information fusion architecture for the IoT is the theme of this paper. Experiments using environmental monitoring sensor data derived from indoor environment are performed for system validation. Through the theoretical analysis and experimental verification, the data processing middleware architecture represents better adaptation to multisensor and multistream application scenarios in the IoT, which improves heterogeneous data utilization value. The data processing middleware based on SOA for the IoT establishes a solid foundation of integration and interaction for diverse networks data among heterogeneous systems in the future, which simplifies the complexity of integration process and improves reusability of components in the system.

1. Introduction

The concept of the Internet of Things (IoT) was firstly derived by the Automatic Identification (Auto-ID) Labs in the Massachusetts Institute of Technology (MIT) in 1999. The Auto-ID Labs simultaneously propose [1] radio frequency identification (RFID) systems that connect devices and transmit information via radio frequency to the Internet in order to achieve intelligent identification and management. To formalize the concept of the “Internet of Things,” the International Telecommunication Union (ITU) released the report of “ITU Internet reports 2005: the Internet of Things [2]” in the World Summit on Information Society (WSIS) held in Tunis in 2005, in which the IoT characteristics, related technical challenges, and future market opportunities were introduced.

ITU pointed out in the report [2], “We are standing at the edge of new times of communication, information and communication technology (ICT) to achieve the objectives have been developed to meet the communication between people, and things, between things connection. The coming

era of ubiquitous Internet of Things make us a new dimension of communication in the world of information and communication technology (shown in Figure 1), any time, any place, connected to anyone, expansion to connect things connected to the Internet of Things.”

With the rapid development of information and communication technology, just a onefold technology cannot satisfy the complex context-aware application requirements, in which resource information has been subject to outside interference, and people want to be able to obtain real-time and real-world information such as diverse sensory data acquisitions and human-computer interaction data acquisitions and ultimately achieve efficient data acquisitions between people and things, people, and things and things.

The IoT applications integrate with IntelliSense recognition technologies, pervasive computing, and ubiquitous networks, which are called the third wave of the information technology revolution following the development of the information industry in the computer and the Internet. The IoT is an important part of the new generation of information

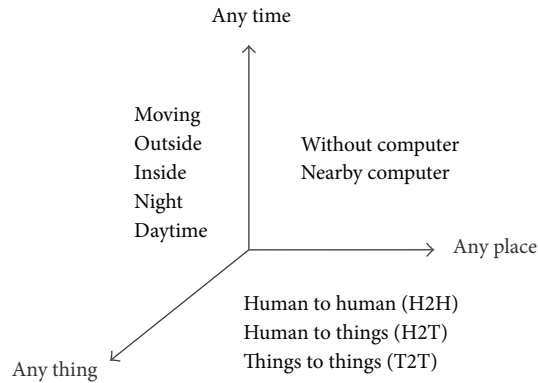


FIGURE 1: Connections in the IoT.

technology. The foundation of the IoT is still the Internet and it is based on the extension and expansion of Internet. The IoT extends its client to anything and any person by the exchange of information and communication through Internet. The IoT connects everything to the Internet.

The IoT incorporates RFID, wireless sensor networks, and ubiquitous terminal equipment as the perception foundation, with a variety of wired or wireless communication and integration of the Internet to achieve the perception data transferred and shared. By leveraging cloud computing and high-performance computing technology for real-time information process, management, and organization, we ultimately offer the upper application a variety of feedback decision-making processes for closed-loop control of the things.

Consequently, the growing popularity of the IoT will inevitably lead to a new wave of development of various industries, such as smart home, intelligent monitoring, smart grid, and other new concepts of things technology. So far, the IoT has been launched as a variety of demonstration applications in different domains (shown in Figure 1), such as intelligent industry [3], intelligent agriculture [4], intelligent logistics [5], intelligent transportation [6], smart grid [7], environmental protection [8], security protection [9], intelligent medical care [10, 11], and smart home [12].

The rest of the paper is organized as follows. In Section 2, the IoT concepts are reviewed. The description of the architecture of the IoT is detailed in Section 3. In Section 4, we propose a middleware framework based on SOA for IoT. We conclude the paper and point out future work in Section 5.

2. Related Work

The IoT will be a promising facility of future network which has self-configuration ability in global dynamic network based on standard and interoperable communication protocols. In the network, all real and virtual items have specific identification and physical sensory data in order to achieve the goal of information sharing through seamless connection of intelligent interface [13, 14]. These intelligent interfaces connect and communicate with users, society, and environment context on the basis of the agreed protocols. It is an extension and expansion of the network based on the

Internet to achieve intelligent identifying, locating, tracking, monitoring, and managing.

From an alternative perspective beyond the initial concept of the IoT and the definitions of it as abovementioned, the IoT is a network connecting things to things for achieving intelligent identification and management of the items in a broad sense; it can be seen as a fusion of the information space and physical space. Through that way, everything is digitized and networked, which results in realizing an efficient information interactive mode between items, items and people, and people and environment. After that, various diversities of information are merged into social networks and integrated into human society in a higher realm. For realization of information fusion in the IoT, the middleware technology is suitable to be adopted as a concrete solution.

Middleware as computer software provides connection of different software components and applications. It consists of a set of enabling services that allow multiple processes running on one or more machines to interact across a network. Atzori et al. [13] summarized the relationship as three visions of the IoT, that is, things-oriented visions, semantic-oriented visions, and Internet-oriented visions. According to the three characteristics, middleware in the IoT shall be able to address things issues and Internet issues, deal with the semantics gap, such as interoperability across heterogeneous devices, context awareness, and device discovery, manage resources constrained embedded devices and scalability, manage large data volumes and privacy, and cope with semantic data and so forth.

Several studies have been published that have explored ways to design middleware for the IoT. In [15], Römer et al. summarized the functions and the nature of the middleware for wireless sensor network. In [16], Wang et al. have reviewed middleware for WSN and a detailed analysis of the approaches and techniques offered by the middleware to meet the requirements of the WSN has been presented. It also discusses generic components of the middleware and reference model of WSN based middleware. In [17], middleware has been surveyed from adaptability perspective in which Sadjadi and McKinley presents taxonomy for adaptive middleware and their application domains and provides details for one of each middleware category. The context-awareness middleware also has been studied. The survey in [18] is based on the architectural aspects and provides taxonomy of the features of the generic context-aware middleware. A survey reported in [19] evaluates several context-aware architectures based on some relevant criteria from ubiquitous or pervasive computing perspective. In [20], Bandyopadhyay et al. provides a survey of middleware system for the IoT.

3. The Architecture of the IoT

Heterogeneous information sources are the most important characters of the IoT. In order to achieve interconnection, intercommunication, and interoperability between heterogeneous information, the future architecture of the IoT needs to be open, layered, and scalable [21]. The IoT architecture is generally divided into four layers that are perception

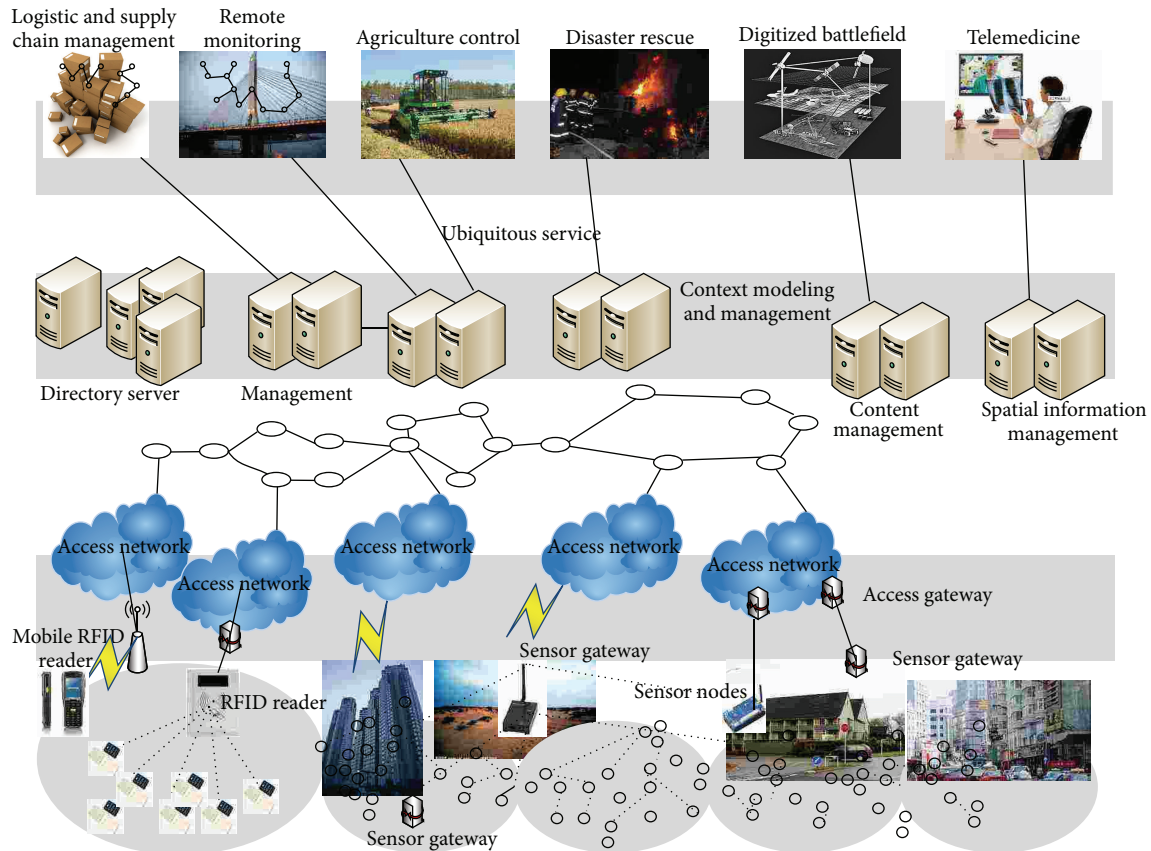


FIGURE 2: Layered framework in the IoT.

layer, network layer, middleware layer, and application layer (shown in Figure 2).

A major part of perception layer is wireless sensor nodes. A generic sensor node aims to take measurements of physical environment [22]. It may be equipped with a variety of devices which can measure various physical attributes such as light, temperature, humidity, barometric pressure, acceleration, acoustics, magnetic field, and carbon dioxide concentration. In addition to the sensors, perception layer also consists of a large amount of information generated equipment, including RFID and positioning systems, and a variety of smart devices, such as smart phones, PDAs, multimedia players, netbooks, and laptops. It can be seen that the diversity of generated information is an emerging and important feature of the IoT.

IPv6 addresses the major defect of the limit on the number of terminal pieces of equipment to access Internet. The main idea of network layer is leveraging the existing Internet as the main dissemination of information by virtue of a variety of wireless accesses. Every wireless access method has its own characteristics and application scenarios. Wi-Fi and other wireless broadband technologies possess broader coverage, faster transmission, reliable high-speed, and lower cost and circumvent the obstacles. The low-speed wireless networks, such as ZigBee, Bluetooth, and infrared low-speed network protocols, are adapted to resource constrained node,

which has the characteristics of the low communication radius, low computing power, and low energy consumption. Mobile communication network will become an effective platform for “a comprehensive, anytime, anywhere.”

Middleware layer tackles the information heterogeneity issues by intelligent interfaces. The functional solutions of middleware layer mainly consist of data storage (database and mass storage technology), heterogeneous data retrieval (search engine), data mining, data security, and privacy protection.

In application layer, traditional Internet has gone through data-centric to people-centric conversion; typical online applications include file transfer, e-mail, the World Wide Web, e-commerce, online gaming, and social networking. In the application of IoT, things or physical world are considered as the center, typical IoT applications covering item tracking, context-aware, intelligent logistics, intelligent transportation, smart grid, and so forth. The IoT application is currently in a period of rapid growth.

4. The Implementation of Middleware Based on SOA for IoT

The IoT research mainly pays more attention to network layer recently, such as the IoT network coding, identification and anticollision technology. However, data processing

infrastructure continues to be overwhelmed by the mass of heterogeneous information from the number of terminals in the IoT. The flexible architecture that is based on SOA for heterogeneous information fusion in the IoT offers the opportunity to employ mitigation measures. It is critical for the ultimate success of the IoT application for better utilization of the integration of a wide range of services from multiple sources and provides more personalized service to businesses or individuals [9].

4.1. Service-Oriented Application Architecture Description. SOA (service-oriented architecture) is a component model and links different functional units (called services) of the application through well-defined interfaces and contracts between these services. Interface is defined by a neutral manner and it should be independent of implementation services, hardware platforms, operating systems, and programming languages. This allows the service to be built in a variety of such systems to interact in a uniform and general way [23]. The service is the basis of the SOA; thereby, they can be applied directly and effectively depending on system and interaction of software agents.

Typically, business operations running in an SOA comprise a number of different components, which are often in an event-driven or asynchronous fashion that reflects the underlying business process needs [24]. In the context of the IoT, original and emerging resources are in the form of services and are opening on Internet. Consequently, the study of SOA-based fusion application technology is of great value [25].

SOA architecture consists of five main parts, depicted as below:

- (1) Consumer: acquires the information from producers' entities that provide services, such as mobile terminals and web clients.
- (2) Application: provides application interfaces or different degrees of loosely coupled services, such as mobile applications, web applications, and rich client.
- (3) Service: the implementation of the entities involved in a specific task, such as data center and enterprise information center.
- (4) Service Support: SOA specific application background support functions, such as security, management, and semantic analysis.
- (5) Producer: an entity to provide specific services or functions.

4.2. The IoT Middleware Design. Inspired by the characteristics of data in the IoT, the design of middleware with the service-oriented architecture was employed in this paper, and integration services, compatible with various types of data and the agreement has been divided. Consequently, this paper presents the basic framework of SOA-based IoT applications as shown in Figure 3. In Figure 3, the three-layer structure of the original SOA is broken down into a five-layer system. Service providers (producers) use of various types of environmental sensing technology. Data processing platform

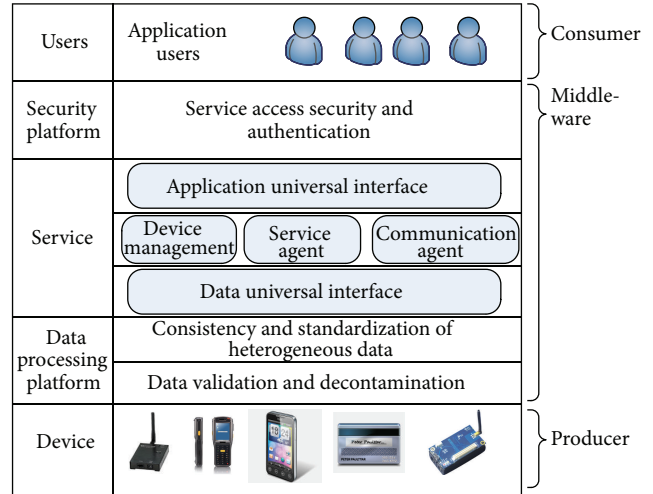


FIGURE 3: The IoT middleware architecture based on SOA.

is responsible for data processing, data filtering, and data integrity. It provides XML scheme for data unification and metadata consistency and standardization of heterogeneous data processing.

Security platform is a security barrier between the service platform and data platform, which is responsible for the safety of the equipment and data. Service layer aims at providing a range of generic interfaces and agency services which are responsible for data parsing in order to coordinate different data formats and are also advantageous for distributed deployment of a variety of databases. The purpose of universal interface is to achieve compatible communication protocols, which are used by different types of users, to perform unified data exchanging with the upper consumers.

The key part of the service layer is to form a bridge between the data processing and the upper application. The service layer also faces different problems that are encountered in the IoT application such as network connection, resource-constrained nodes, and different application platform. Because the underlying device is extremely rich in the IoT, the SOA system to provide network services needs to consider the problem of transmission delay and resource scheduling and network services need to provide a variety of routing or delay tolerant network technology to deal with. SOA systems also need a balanced scheduling algorithm and balanced network resources. Different application platforms require more generic SOA system design patterns; we will first consider the standard between different devices and the upper users between different access platforms.

As it can be seen from Figure 3, the basic framework of SOA application is on the basis of the data stream generated by the perceive network from the physical world, which is with the basic physical properties of the world from the underlying environmental sensing. In SOA architecture, these vast amounts of real context-aware data form the basis of the entire application.

Since heterogeneous data processing is inevitably linked to the IoT middleware architecture based on SOA, a concrete

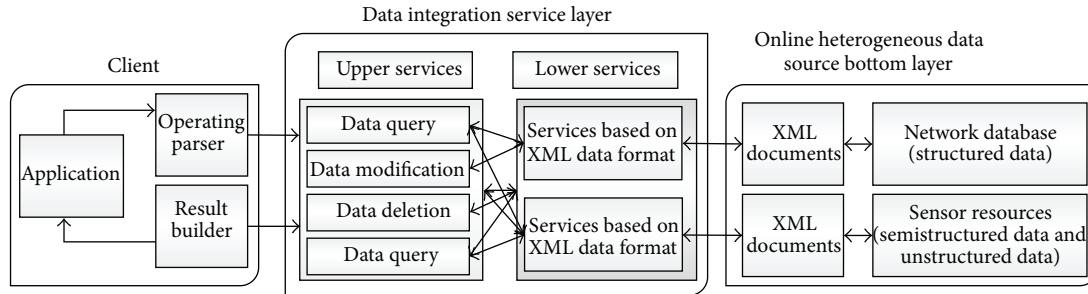


FIGURE 4: The IoT data integration middleware based on SOA.

solution is proposed for the metadata integration of heterogeneous data which is as shown in Figure 4. This architecture is divided into three basic processes, which from top to bottom are the client application layer, the data integration layer, and the IoT heterogeneous data sources, as shown in Figure 4.

The client applications include users unified access interface for data manipulation, which can be a specific application or a web browser.

Data integration service layer [26] is the core of the architecture and also the key to heterogeneous data integration. In order to increase the intelligence and scalability of architecture and alleviate the burden on users, we design a structure that contains the upper and lower levels of service. Metadata format vary greatly since which are grounded in heterogeneous sensor sources [27]. To circumvent this obstacle, we express various types of data into XML format and set up rules to make operations on the metadata. Consequently, we first converted the heterogeneous data in a unified XML format, on this basis, and then created the underlying data integration services layer. The upper layer services built on top of the lower layer services and the underlying services are developed in an XML document based on the underlying heterogeneous data sources.

Services of lower levels achieve four data access functions: add, modify, query, and delete and the upper service function extracts the same functionality from the underlying service to the same service according to the data then forms the integrated data service function. Application layer call corresponding upper services according to the operational requirements, and then underlying data manipulation is specified by the upper service based on data parameters from the client calls; thus, when the underlying heterogeneous data source changes, we simply update the underlying service and map to the upper applications rather than make any changes to upper layers. Data integration implementation process is completely transparent to the user, which is compatible and interoperable in different systems.

4.3. Evaluation. In order to test the IoT middleware architecture based on SOA, we set up an indoor temperature monitoring system in practical environment. We deploy 30 sensor nodes in three rooms, to monitor the indoor temperatures. After collection of the sensor data, the data is delivered by a wireless multihop network from sensor nodes to a base station which is connected to a cluster.

In experiments, clusters include 4 common PCs that are 2-core 2.8 GHz desktop computer with 2 GB RAM and Ubuntu operating system for the runtime environment. Although they are generic computers, they satisfy the requirements.

In the paper, the sensor nodes are IRIS nodes. IRIS nodes are produced by Crossbow Technology Inc. These sensors are based on the Atmegall281 microprocessing chip and a RF230 RF chip which are working at 2.4 GHz and supporting the IEEE 802.15.4 communication protocol. The nodes have three times radio range and twice the program memory of MICA Motes and outdoor line-of-sight tests have a range as great as 500 meters between nodes without amplification. The IRIS not only has a longer transmission distance but also has ultralow power consumption and a longer battery life.

In the test, equipment used is shown in Figure 5 and the portal of the IoT system for environment monitoring is shown in Figure 6. Some example of the collected sensory data is as shown in Table 1. The monitoring system continues evaluating for three months and collects about five million of the sensor data items. During the test period, we have supplemented and changed several types of the sensors without interrupting the system. When we add the new equipment, some new profiles will be added to the middleware framework while the system is still running.

We can draw some conclusions from the result of the experiment.

First, the SOA-based data processing middleware architecture represents better adaptation to the IoT multisensor and multistream application scenarios, which improve the heterogeneous data reusability and utilization value.

Second, the experiment result demonstrates the decoupling power of middleware which makes it easier to establish a unified heterogeneous information processing platform for a diversity of applications in the IoT.

Third, the distributed deployment of middleware brings better performance optimization and achieves better load balancing in the cluster.

5. Conclusion

This paper discusses the development of the concept of the IoT and gives a detailed description of the architecture of the IoT. Based on characteristics of the architecture and challenges of information fusion in the IoT, the paper designs a middleware platform based on SOA architecture for the

TABLE 1: Some examples of the collected sensory data.

Node ID	Parent	Board_ID	Voltage	Humid	hum temp	pr temp	Press	accel_x	accel_y	accel_z	Time
3	4	133	2988	27	23	22.13008	939.5078	4062	125	994	33:05.8
9	0	133	2996	27	24	22.42129	940.9485	19	140	1010	32:57.8
6	9	133	2389	-4	-39	17.67822	931.6279	4086	64	1026	27:08.9
8	0	133	2345	-4	-39	19.73633	943.171	57	84	3076	41:28.3
10	0	133	1950	-4	-39	11.34609	938.7853	58	4094	3142	16:08.9
2	0	133	1911	1	-39	19.94141	937.0197	209	42	3119	32:47.7
5	1	133	1917	15	-39	16.5043	942.0411	33	55	3111	07:09.5
1	9	133	2728	53	20	18.78281	942.9518	3847	104	1018	29:46.8
7	9	133	2981	28	23	21.90244	940.4976	4071	83	994	30:14.3
4	0	133	3017	26	24	21.84063	941.601	4088	106	1017	33:07.9
0	9	133	3025	35	22	20.2375	933.5154	4079	136	1010	10:37.8

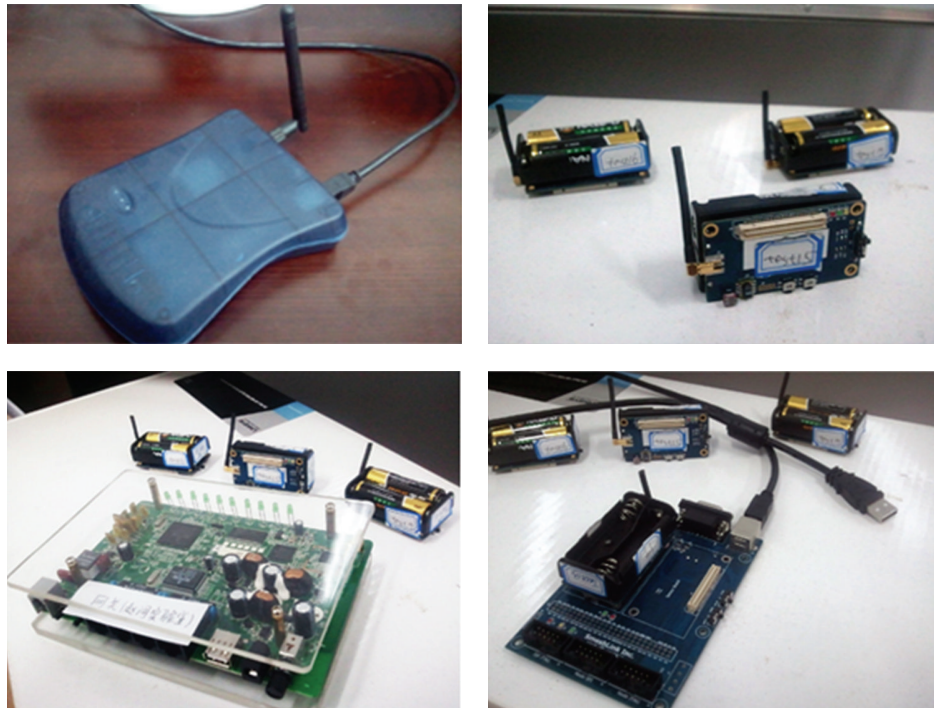


FIGURE 5: The equipment used in the demo.

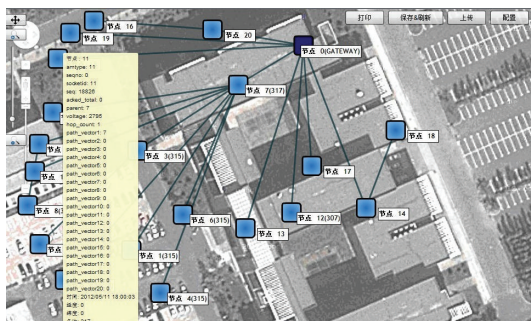


FIGURE 6: The portal of the IoT system for environment monitoring.

integration of multisource heterogeneous information. After that, we use the SOA data processing middleware to build an environmental monitoring system for validation verifying. Through theoretical analysis and experimental verification, the SOA pattern-based processing middleware architecture design is better adapted to the IoT multisensor and multi-stream application scenarios, which improve the sensing data utilization value. The SOA data processing middleware has laid a solid foundation for data integration and interaction between different networking systems, simplifying the complexity of the system integration process and improving the reuse of components in the future. In order to achieve better

interaction between the different large-scale IoT applications, the criteria with regard to unified data format are widely considered to be made for coordination of different systems in relevant international organizations, research institutions, and enterprises.

Conflict of Interests

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

Acknowledgments

This work is funded by the European Framework Program (FP7) under Grant no. FP7-PEOPLE-2011-IRSES, the National Natural Science Foundation of China under Grant nos. 61073009 and 60873235 and 61103197, the National High Technology R&D Program 863 of China under Grant no. 2011AA010101, the National Key Technology R&D Program of China under Grant no. SQ2013GX11E00316-F03, the National Science and Technology Major Projects of China under Grant nos. SinoProbe-09-01-03 and 2012ZX01039-004-04-3, the Key Science Technology Program of Jilin Province of China under Grant no. 2011ZDGG007, the Key Scientific and Technological Project of Jilin Province of China under Grant no. 20150204035GX, and the Fundamental Research Funds for Central Universities of China under Grant no. JCKY-QKJC46.

References

- [1] S. Sarma, D. L. Brock, and K. Ashton, "The networked physical world—proposals for engineering the next generation of computing, commerce & automatic identification," *Auto-ID Centre White paper*, 2000, <http://www.foxner.com>.
- [2] International Telecommunication Union, *ITU Internet Reports 2005: The Internet of Things*, International Telecommunication Union, 2005.
- [3] J. Li, Y. Zhang, D. Zhou, H. Zhang, and H. Xu, "Design and application of a new IOT reader," in *Proceedings of the 2nd International Conference on Information Science and Engineering (ICISE '10)*, pp. 1944–1947, IEEE, Hangzhou, China, December 2010.
- [4] D. Yan-E, "Design of intelligent agriculture management information system based on IoT," in *Proceedings of the 4th International Conference on Intelligent Computation Technology and Automation (ICICTA '11)*, pp. 1045–1049, March 2011.
- [5] X. Wang, W. Li, Y. Zhong, and W. Zhao, "Research on cloud logistics-based one-stop service platform for logistics center," in *Proceedings of the IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD '12)*, pp. 558–563, Wuhan, China, May 2012.
- [6] L. Foschini, T. Taleb, A. Corradi, and D. Bottazzi, "M2M-based metropolitan platform for IMS-enabled road traffic management in IoT," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 50–57, 2011.
- [7] H. Dahai, Z. Jie, Z. Yongjun, and G. Wanyi, "Convergence of sensor networks/internet of things and power grid information network at aggregation layer," in *International Conference on Power System Technology (POWERCON '10)*, pp. 1–6, Hangzhou, China, 2010.
- [8] H. Wang, T. Zhang, Y. Quan, and R. Dong, "Research on the framework of the environmental internet of things," *International Journal of Sustainable Development and World Ecology*, vol. 20, no. 3, pp. 199–204, 2013.
- [9] C. Du and S. Zhu, "Research on urban public safety emergency management early warning system based on technologies for the Internet of things," *Procedia Engineering*, vol. 45, pp. 748–754, 2012.
- [10] L. Dongxin and L. Tao, "The application of IOT in medical system," in *Proceedings of the IEEE International Symposium on IT in Medicine and Education (ITME '11)*, pp. 272–275, December 2011.
- [11] A. J. Jara, M. A. Zamora, and A. F. G. Skarmeta, "An architecture based on internet of things to support mobility and security in medical environments," in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC '10)*, pp. 1–5, IEEE, Las Vegas, Nev, USA, January 2010.
- [12] M. Darianian and M. P. Michael, "Smart home mobile RFID-based internet-of-things systems and services," in *Proceedings of the International Conference on Advanced Computer Theory and Engineering (ICACTE '08)*, pp. 116–120, December 2008.
- [13] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [14] L. Xu, W. He, and S. Li, "Internet of things in industries: a survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [15] K. Römer, O. Kasten, and F. Mattern, "Middleware challenges for wireless sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 4, pp. 59–61, 2002.
- [16] M.-M. Wang, J.-N. Cao, J. Li, and S. K. Dasi, "Middleware for wireless sensor networks: a survey," *Journal of Computer Science and Technology*, vol. 23, no. 3, pp. 305–326, 2008.
- [17] S. M. Sadjadi and P. K. McKinley, "A survey of adaptive middleware," Report MSU-CSE-03-35, Michigan State University, 2003.
- [18] K. E. Kjær, "A survey of context-aware middleware," in *Proceedings of the IASTED International Conference on Software Engineering (SE '07)*, pp. 148–155, February 2007.
- [19] M. Miraoui, C. Tadj, and C. B. Amar, "Architectural survey of context-aware systems in pervasive computing environment," *Ubiquitous Computing and Communication Journal*, vol. 3, no. 3, pp. 1–9, 2008.
- [20] S. Bandyopadhyay, M. Sengupta, S. Maiti, and S. Dutta, "A survey of middleware for internet of things," in *Recent Trends in Wireless and Mobile Networks*, pp. 288–296, Springer, Berlin, Germany, 2011.
- [21] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
- [22] M. H. Alizai, H. Wirtz, B. Kirchen, and K. Wehrle, "Portable wireless-networking protocol evaluation," *Journal of Network and Computer Applications*, vol. 36, no. 4, pp. 1230–1242, 2013.
- [23] M. Kovatsch, M. Lanter, and S. Duquennoy, "Actinium: a RESTful runtime container for scriptable internet of things applications," in *Proceedings of the 3rd International Conference on the Internet of Things (IOT '12)*, pp. 135–142, October 2012.

- [24] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Sensing as a service model for smart cities supported by internet of things," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 1, pp. 81–93, 2014.
- [25] D. Schall, F. Skopik, and S. Dustdar, "Expert discovery and interactions in mixed service-oriented systems," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 233–245, 2012.
- [26] J. Zhou, L. Hu, J. Chu, H. Lu, F. Wang, and K. Zhao, "Feature selection from incomplete multi-Sensor information system based on positive approximation in rough set theory," *Sensor Letters*, vol. 11, no. 5, pp. 974–981, 2013.
- [27] J. Zhou, L. Hu, F. Wang, H. Lu, and K. Zhao, "An efficient multidimensional fusion algorithm for iot data based on partitioning," *Tsinghua Science and Technology*, vol. 18, no. 4, Article ID 6574675, pp. 369–378, 2013.