

Advanced Mathematics and Numerical Modeling of IoT

Guest Editors: Young-Sik Jeong, Mohammad S. Obaidat, Jianhua Ma,
and Laurence T. Yang





Advanced Mathematics and Numerical Modeling of IoT

Journal of Applied Mathematics

Advanced Mathematics and Numerical Modeling of IoT

Guest Editors: Young-Sik Jeong, Mohammad S. Obaidat,
Jianhua Ma, and Laurence T. Yang



Copyright © 2015 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in "Journal of Applied Mathematics." 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

- Saeid Abbasbandy, Iran
Mina B. Abd-El-Malek, Egypt
Mohamed A. Abdou, Egypt
Subhas Abel, India
Mostafa Adimy, France
Carlos J. S. Alves, Portugal
Mohamad Alwash, USA
Igor Andrianov, Germany
Sabri Arik, Turkey
Francis T.K. Au, Hong Kong
Olivier Bahn, Canada
Roberto Barrio, Spain
Alfredo Bellen, Italy
Jafar Biazar, Iran
Hester Bijl, The Netherlands
Anjan Biswas, Saudi Arabia
Stephane P.A. Bordas, USA
James Robert Buchanan, USA
Alberto Cabada, Spain
Xiao Chuan Cai, USA
Jinde Cao, China
Alexandre Carvalho, Brazil
Song Cen, China
Qianshun S. Chang, China
Tai-Ping Chang, Taiwan
Shih-sen Chang, China
Rushan Chen, China
Xinfu Chen, USA
Ke Chen, UK
Eric Cheng, Hong Kong
Francisco Chiclana, UK
Jen-Tzung Chien, Taiwan
C. S. Chien, Taiwan
Han H. Choi, Republic of Korea
Tin-Tai Chow, China
M. S. H. Chowdhury, Malaysia
Carlos Conca, Chile
Vitor Costa, Portugal
Livija Cveticanin, Serbia
Eric de Sturler, USA
Orazio Descalzi, Chile
Kai Diethelm, Germany
Vit Dolejsi, Czech Republic
Bo-Qing Dong, China
Magdy A. Ezzat, Egypt
- Meng Fan, China
Ya Ping Fang, China
Antonio J. M. Ferreira, Portugal
Michel Fliess, France
M. A. Fontelos, Spain
Huijun Gao, China
Bernard J. Geurts, The Netherlands
Jamshid Ghaboussi, USA
Pablo González-Vera, Spain
Laurent Gosse, Italy
K. S. Govinder, South Africa
Jose L. Gracia, Spain
Yuantong Gu, Australia
Zhihong GUAN, China
Nicola Guglielmi, Italy
Frederico G. Guimarães, Brazil
Vijay Gupta, India
Bo Han, China
Maoan Han, China
Pierre Hansen, Canada
Ferenc Hartung, Hungary
Xiaoqiao He, Hong Kong
Luis Javier Herrera, Spain
J. Hoenderkamp, The Netherlands
Ying Hu, France
Ning Hu, Japan
Zhilong L. Huang, China
Kazufumi Ito, USA
Takeshi Iwamoto, Japan
George Jaiani, Georgia
Zhongxiao Jia, China
Tarun Kant, India
Ido Kanter, Israel
Abdul Hamid Kara, South Africa
Hamid Reza Karimi, Norway
Jae-Wook Kim, UK
Jong Hae Kim, Republic of Korea
Kazutake Komori, Japan
Fanrong Kong, USA
Vadim . Krysko, Russia
Jin L. Kuang, Singapore
Miroslaw Lachowicz, Poland
Hak-Keung Lam, UK
Tak-Wah Lam, Hong Kong
PGL Leach, Cyprus
- Yongkun Li, China
Wan-Tong Li, China
J. Liang, China
Ching-Jong Liao, Taiwan
Chong Lin, China
Mingzhu Liu, China
Chein-Shan Liu, Taiwan
Kang Liu, USA
Yansheng Liu, China
Fawang Liu, Australia
Shutian Liu, China
Zhijun Liu, China
Julián López-Gómez, Spain
Shiping Lu, China
Gert Lube, Germany
Nazim Idrisoglu Mahmudov, Turkey
Oluwole Daniel Makinde, South Africa
Francisco J. Marcellán, Spain
Guiomar Martín-Herrán, Spain
Nicola Mastronardi, Italy
Michael McAleer, The Netherlands
Stephane Metens, France
Michael Meylan, Australia
Alain Miranville, France
Ram N. Mohapatra, USA
Jaime E. Munoz Rivera, Brazil
Javier Murillo, Spain
Roberto Natalini, Italy
Srinivasan Natesan, India
Jiri Nedoma, Czech Republic
Jianlei Niu, Hong Kong
Roger Ohayon, France
Javier Oliver, Spain
Donal O'Regan, Ireland
Martin Ostoja-Starzewski, USA
Turgut Öziş, Turkey
Claudio Padra, Argentina
Reinaldo Martinez Palhares, Brazil
Francesco Pellicano, Italy
Juan Manuel Peña, Spain
Ricardo Perera, Spain
Malgorzata Peszynska, USA
James F. Peters, Canada
Mark A. Petersen, South Africa
Miodrag Petkovic, Serbia

Vu Ngoc Phat, Vietnam	Abdel-Maksoud A. Soliman, Egypt	Qing-Wen Wang, China
Andrew Pickering, Spain	Xinyu Song, China	Guangchen Wang, China
Hector Pomares, Spain	Qiankun Song, China	Junjie Wei, China
Maurizio Porfiri, USA	Yuri N. Sotskov, Belarus	Li Weili, China
Mario Primicerio, Italy	Peter J. C. Spreij, The Netherlands	Martin Weiser, Germany
Morteza Rafei, The Netherlands	Niclas Strömberg, Sweden	Frank Werner, Germany
Roberto Renò, Italy	Ray KL Su, Hong Kong	Shanhe Wu, China
Jacek Rokicki, Poland	Jitao Sun, China	Dongmei Xiao, China
Dirk Roose, Belgium	Wenyu Sun, China	Gongnan Xie, China
Carla Roque, Portugal	XianHua Tang, China	Yuesheng Xu, USA
Debasish Roy, India	Alexander Timokha, Norway	Suh-Yuh Yang, Taiwan
Samir H. Saker, Egypt	Mariano Torrisi, Italy	Bo Yu, China
Marcelo A. Savi, Brazil	Jung-Fa Tsai, Taiwan	Jinyun Yuan, Brazil
Wolfgang Schmidt, Germany	Ch Tsitouras, Greece	Alejandro Zarzo, Spain
Eckart Schnack, Germany	Kuppapalle Vajravelu, USA	Guisheng Zhai, Japan
Mehmet Sezer, Turkey	Alvaro Valencia, Chile	Jianming Zhan, China
Naseer Shahzad, Saudi Arabia	Erik Van Vleck, USA	Zhijia Zhang, China
Fatemeh Shakeri, Iran	Ezio Venturino, Italy	Jingxin Zhang, Australia
Jian Hua Shen, China	Jesus Vigo-Aguiar, Spain	Shan Zhao, USA
Hui-Shen Shen, China	Michael N. Vrahatis, Greece	Chongbin Zhao, Australia
Fernando Simões, Portugal	Baolin Wang, China	Renat Zhdanov, USA
Theodore E. Simos, Greece	Mingxin Wang, China	Hongping Zhu, China

Contents

Advanced Mathematics and Numerical Modeling of IoT, Young-Sik Jeong, Mohammad S. Obaidat, Jianhua Ma, and Laurence T. Yang
Volume 2015, Article ID 824891, 5 pages

Investigation Methodology of a Virtual Desktop Infrastructure for IoT, Doowon Jeong, Jungheum Park, Sangjin Lee, and Chulhoon Kang
Volume 2015, Article ID 689870, 10 pages

Fall-Detection Algorithm Using 3-Axis Acceleration: Combination with Simple Threshold and Hidden Markov Model, Dongha Lim, Chulho Park, Nam Ho Kim, Sang-Hoon Kim, and Yun Seop Yu
Volume 2014, Article ID 896030, 8 pages

Linear SVM-Based Android Malware Detection for Reliable IoT Services, Hyo-Sik Ham, Hwan-Hee Kim, Myung-Sup Kim, and Mi-Jung Choi
Volume 2014, Article ID 594501, 10 pages

SCondi: A Smart Context Distribution Framework Based on a Messaging Service for the Internet of Things, Jongmoon Park and Myung-Joon Lee
Volume 2014, Article ID 271817, 8 pages

A Prediction System Using a P2P Overlay Network for a Bus Arrival System, Ssu-Hsuan Lu and Yu-Wei Chan
Volume 2014, Article ID 792029, 7 pages

Secure Collaborative Key Management for Dynamic Groups in Mobile Networks, Sukin Kang, Cheongmin Ji, and Manpyo Hong
Volume 2014, Article ID 601625, 10 pages

Modeling and Implementing Two-Stage AdaBoost for Real-Time Vehicle License Plate Detection, Moon Kyou Song and Md. Mostafa Kamal Sarker
Volume 2014, Article ID 697658, 8 pages

Performance Improvement Based Authentication Protocol for Intervessel Traffic Service Data Exchange Format Protocol Based on U-Navigation System in WoT Environment, Byunggil Lee and Namje Park
Volume 2014, Article ID 734768, 7 pages

A Rational Threshold Signature Model and Protocol Based on Different Permissions, Bojun Wang, Cheng Cai, and Quan Zhou
Volume 2014, Article ID 176085, 9 pages

Development of the Korean Spine Database and Automatic Surface Mesh Intersection Algorithm for Constructing e-Spine Simulator, Dongmin Seo, Hanmin Jung, Won-Kyung Sung, and Dukyun Nam
Volume 2014, Article ID 471756, 11 pages

Building a Smooth Medical Service for Operating Room Using RFID Technologies, Lun-Ping Hung, Hsin-Ke Lu, Ching-Sheng Wang, and Ding-Jung Chiang
Volume 2014, Article ID 984721, 11 pages

Taxonomy and Evaluations of Low-Power Listening Protocols for Machine-to-Machine Networks,
Kwang-il Hwang and Sung-Hyun Yoon
Volume 2014, Article ID 138571, 12 pages

Task Balanced Workflow Scheduling Technique considering Task Processing Rate in Spot Market,
Daeyong Jung, JongBeom Lim, JoonMin Gil, Eunyoung Lee, and Heonchang Yu
Volume 2014, Article ID 237960, 10 pages

Mathematical Modeling of a Multilayered Drift-Stabilization Method for Micro-UAVs Using Inertial Navigation Unit Sensor, Hyeok-June Jeong, Myunggwon Hwang, Hanmin Jung, and Young-guk Ha
Volume 2014, Article ID 747134, 11 pages

Towards Self-Awareness Privacy Protection for Internet of Things Data Collection,
Kok-Seng Wong and Myung Ho Kim
Volume 2014, Article ID 827959, 9 pages

A Study on Intelligent User-Centric Logistics Service Model Using Ontology, Saraswathi Sivamani, Kyunghun Kwak, and Yongyun Cho
Volume 2014, Article ID 162838, 10 pages

Numeric Analysis for Relationship-Aware Scalable Streaming Scheme, Heung Ki Lee, Jaehee Jung, Kyung Jin Ahn, Hwa-Young Jeong, and Gangman Yi
Volume 2014, Article ID 195781, 12 pages

Medical Image Segmentation for Mobile Electronic Patient Charts Using Numerical Modeling of IoT, Seung-Hoon Chae, Daesung Moon, Deok Gyu Lee, and Sung Bum Pan
Volume 2014, Article ID 815039, 8 pages

REST-MapReduce: An Integrated Interface but Differentiated Service, Jong-Hyuk Park, Hwa-Young Jeong, Young-Sik Jeong, and Min Choi
Volume 2014, Article ID 170723, 10 pages

Estimated Interval-Based Checkpointing (EIC) on Spot Instances in Cloud Computing, Daeyong Jung, JongBeom Lim, Heonchang Yu, and Taeweon Suh
Volume 2014, Article ID 217547, 12 pages

Whitelists Based Multiple Filtering Techniques in SCADA Sensor Networks, DongHo Kang, ByoungKoo Kim, JungChan Na, and KyoungSon Jhang
Volume 2014, Article ID 597697, 7 pages

TSMC: A Novel Approach for Live Virtual Machine Migration, Jiaying Song, Weidong Liu, Feiran Yin, and Chao Gao
Volume 2014, Article ID 297127, 7 pages

Grid-PPPS: A Skyline Method for Efficiently Handling Top-?? Queries in Internet of Things, Sun-Young Ihm, Aziz Nasridinov, and Young-Ho Park
Volume 2014, Article ID 401618, 10 pages

Contents

Analysis and Enhancement of IEEE 802.15.4e DSME Beacon Scheduling Model,

Kwang-il Hwang and Sung-wook Nam
Volume 2014, Article ID 934610, 15 pages

Optimization of High-Speed Train Control Strategy for Traction Energy Saving Using an Improved Genetic Algorithm, Ruidan Su, Qianrong Gu, and Tao Wen

Volume 2014, Article ID 507308, 7 pages

Botnet Detection Using Support Vector Machines with Artificial Fish Swarm Algorithm,

Kuan-Cheng Lin, Sih-Yang Chen, and Jason C. Hung
Volume 2014, Article ID 986428, 9 pages

Adaptive Failure Identification for Healthcare Risk Analysis and Its Application on E-Healthcare,

Kuo-Chung Chu and Lun-Ping Hung
Volume 2014, Article ID 865241, 17 pages

Energy-Efficient Probabilistic Routing Algorithm for Internet of Things, Sang-Hyun Park,

Seungryoung Cho, and Jung-Ryun Lee
Volume 2014, Article ID 213106, 7 pages

Usability Analysis of Collision Avoidance System in Vehicle-to-Vehicle Communication Environment,

Hong Cho, Gyoung-Eun Kim, and Byeong-Woo Kim
Volume 2014, Article ID 951214, 10 pages

An Efficient and Secure *m*-IPS Scheme of Mobile Devices for Human-Centric Computing,

Young-Sik Jeong, Jae Dong Lee, Jeong-Bae Lee, Jai-Jin Jung, and Jong Hyuk Park
Volume 2014, Article ID 198580, 8 pages

Exponential Stability for Impulsive Stochastic Nonlinear Network Systems with Time Delay,

Lanping Chen, Zhengzhi Han, and Zhenghua Ma
Volume 2014, Article ID 787568, 5 pages

Ubiquitous Health Management System with Watch-Type Monitoring Device for Dementia Patients,

Dongmin Shin, Dongil Shin, and Dongkyoo Shin
Volume 2014, Article ID 878741, 8 pages

Editorial

Advanced Mathematics and Numerical Modeling of IoT

Young-Sik Jeong,¹ Mohammad S. Obaidat,² Jianhua Ma,³ and Laurence T. Yang⁴

¹*Department of Multimedia Engineering, Dongguk University, Seoul 100-715, Republic of Korea*

²*Department of Computer Science and Software Engineering, Monmouth University, West Long Branch, NJ 07764-1898, USA*

³*Faculty of Computer and Information Sciences, Hosei University, Tokyo 102-8160, Japan*

⁴*Department of Computer Science, St. Francis Xavier University, P.O. Box 5000, Antigonish, NS, Canada*

Correspondence should be addressed to Young-Sik Jeong; ysjeong@dongguk.edu

Received 13 October 2014; Accepted 13 October 2014

Copyright © 2015 Young-Sik Jeong 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.

Recent advances in the Internet of Things (IoT) have posed great challenges to computer science and engineering. Internet of Things systems manage huge numbers of heterogeneous sensors and/or mobile devices which continuously monitor the states of real-world objects, and most data are generated automatically through mobile networking environments. Internet of Things frameworks might help support the interaction between “things” and allow for more complex structures like Distributed/Grid/Cloud computing and the development of Distributed/Grid/Cloud applications. Currently, some Internet of Things frameworks seem to focus on real-time data logging solutions, which are offering some basis to work with many “things.” Future developments might lead to specific software development environments to create the software to work with the hardware used in the Internet of Things [1–3]. IoT is also a novel paradigm that is rapidly gaining in the scenario of wireless sensor networks and wireless telecommunications. The basic idea of this concept is the pervasive presence around our life style of a variety of things or objects [2, 4, 5].

The advanced mathematics and numerical modeling of IoT is being driven by and used in a wide range of academic, research, and commercial application areas. This use is producing important new practical experience in a variety of different problem domains in each of these areas. There are also new computational methods in these research fields. As an enabler, this technology is leading to a rapid growth in both scientific and information applications that will, in turn, enable additional requirements for the advanced

mathematics and numerical modeling of Internet of Things to be identified. These will impact academia, business, and education [1, 3, 5].

This special issue aims to provide advanced theory and application for researchers and practitioners to contribute original research and review articles that present the state-of-the-art research outcomes, practical results, latest findings, and future evolutions of mathematics in IoT applications.

We have received many manuscripts and the published manuscripts with high quality were finally selected for this special issue. Each manuscript selected was blindly reviewed by at least three reviewers consisting of guest editors and external reviewers. We present a brief overview of each manuscript in the following.

Recent advances in mathematics and numerical modeling of Internet of Things have created new topics of interest including the following: (1) mathematical and numerical modeling of smart city including smartphone real-world and mobile networks, (2) optimization methods, mathematics modeling for smart Grid with Cloud computing, (3) numerical analysis for security and emergencies in IoT, (4) methods for improving efficiency or accuracy of M2M applications and vehicle autodiagnosis, (5) adaptive and dynamic algorithms for home automation and e-health applications, (6) computational models of communication mechanisms for mobile networks in IoT, and (7) advanced modeling for IoT applications as mobile/vehicle ad hoc networks and mobile sensor networks in CPS (cyber physical system).

In these several topics of the advanced mathematics and numerical modeling of IoT (Internet of Things), some articles proposed the following.

D. Lim et al. proposed a fall-detection algorithm using 3-axis acceleration. The fall-feature parameters, calculated from the 3-axis acceleration, are applied to a simple threshold method. Then, the falls that are determined from the simple threshold are applied to the HMM to distinguish between falls and ADLs. The results from a simple threshold, HMM, and the combination of the simple method and HMM are compared and analyzed.

H.-S. Ham et al. applied a linear support vector machine (SVM) to detect Android malware and compare the malware detection performance of SVM with that of other machine learning classifiers. Through experimental validation, they show that the SVM outperforms other machine learning classifiers.

J. Park and M.-J. Lee presented a context distribution framework named SCondi utilizing the messaging service which supports MQTT—an OASIS standard IoT messaging protocol. SCondi provides the notion of context channel as a core feature to support efficient and reliable mechanism for distributing huge context information in the IoT environment. The context channel provides a pluggable filter mechanism that supports effective extraction, tailoring, authentication, and security of information.

S.-H. Lu and Y.-W. Chan proposed a method of combining a peer-to-peer (P2P) overlay network and WSN to develop a bus arrival time prediction system. A P2P overlay network is added to traditional prediction systems to allow for real-time data. Each bus is installed with a sensor, and each bus stop can receive data sent from sensors. Due to the distance limitation of sensors, the sensors on buses and the data-receiving device of a bus station form a single WSN environment. All bus stations and station termini are connected to form a P2P overlay network, which is used to transmit real-time bus information and predict bus arrival times. Through the WSN technology, bus stations retrieve data from buses and transmit these data to subsequent bus stations to estimate the bus arrival times. This approach can be a powerful tool for monitoring and predicting traffic conditions.

S. Kang et al. proposed a group key sharing scheme and efficient rekeying methods for frequent membership changes from network dynamics. The proposed method enables the group members to simply establish a group key and provide high flexibility for dynamic group changes such as member joining or leaving and group merging or partition. They conduct mathematical evaluation with other group key management protocols and finally prove its security by demonstrating group key secrecy, backward and forward secrecy, key independence, and implicit key authentication under the decisional Diffie-Hellman (DDH) assumption.

M. K. Song and M. K. Sarker proposed automatic detection of car LPs via image processing techniques based on classifier or machine learning algorithms. In this paper, they propose a real-time and robust method for LPD systems using the two-stage adaptive boosting (AdaBoost) algorithm combined with different image preprocessing techniques.

Haar-like features are used to compute and select features from LP images. The AdaBoost algorithm is used to classify parts of an image within a search window by a trained strong classifier as either LP or non-LP. Adaptive thresholding is used for the image preprocessing method applied to those images that are of insufficient quality for LPD. This method is of a faster speed and higher accuracy than most of the existing methods used in LPD. Experimental results demonstrate that the average LPD rate is 98.38% and the computational time is approximately 49 ms.

B. Lee and N. Park, International Association of Lighthouse Authorities (IALA), are developing the standard inter-system VTS exchange format (IVEF) protocol for exchange of navigation and vessel information between VTS systems and between VTS and vessels. VTS (vessel traffic system) is an important marine traffic monitoring system which is designed to improve the safety and efficiency of navigation and the protection of the marine environment. And the demand of inter-VTS networking has been increased for realization of e-navigation as shore side collaboration for maritime safety. And IVEF (inter-VTS data exchange format) for inter-VTS network has become a hot research topic of VTS system. Currently, the IVEF developed by the International Association of Lighthouse Authorities (IALA) does not include any highly trusted certification technology for the connectors. The output of standardization is distributed as the IALA recommendation V-145, and the protocol is implemented with an open source. The IVEF open source, however, is the code used to check the functions of standard protocols. It is too slow to be used in the field and requires a large memory. And the vessel traffic information requires high security since it is highly protected by the countries. Therefore, this paper suggested the authentication protocol to increase the security of the VTS systems using the main certification server and IVEF.

B. Wang et al. developed a novel model and protocol used in some specific scenarios, in which the participants of multiple groups with different permissions can finish the signature together. They applied the secret sharing scheme based on difference equation to the private key distribution phase and secret reconstruction phrase of their threshold signature scheme. In addition, their scheme can achieve the signature success because of the punishment strategy of the repeated rational secret sharing. Besides, the bit commitment and verification method used to detect players' cheating behavior acts as a contributing factor to prevent the internal fraud.

D. Seo et al. presented the Korean spine database and automatic surface mesh intersection algorithm to construct e-spine. To date, the Korean spine database has collected spine data from 77 cadavers and 298 patients. The spine data consists of 2D images from CT, MRI, or X-ray, 3D shapes, geometry data, and property data. The volume and quality of the Korean spine database are now the world's highest ones. In addition, their triangular surface mesh intersection algorithm automatically remeshes the spine-implant intersection model to make it valid for finite element analysis (FEA). This makes it possible to run the FEA using the spine-implant mesh model without any manual effort.

L.-P. Hung et al. proposed a new framework using radio frequency identification (RFID) technology for a mobilized surgical process monitoring system. Through the active tag, an application management system used before, during, and after the surgical processes has been proposed. The concept of signal level matrix, SLM, was proposed to accurately identify patients and dynamically track patients' location. By updating patient's information real time, the preprocessing time needed for various tasks and incomplete transfers among departments can be reduced, the medical resources can be effectively used, unnecessary medical disputes can be reduced, and more comprehensive health care environment can be provided. The feasibility and effectiveness of their proposed system are demonstrated with a number of experimental results.

K. Hwang and S.-H. Yoon provided a well-defined taxonomy of low-power listening protocols by examining in detail the existing low-power sensor network protocols and evaluation results. It will also be very useful for helping M2M designers understand specific features of low-power media access control protocols as they design new M2M networks.

D. Jung et al. proposed the workflow scheduling scheme that reduces the out-of-bid situation. Consequently, the total task completion time is decreased. The simulation results reveal that, compared to various instance types, their scheme achieves performance improvements in terms of an average combined metric of 12.76% over workflow scheme without considering the processing rate. However, the cost in their scheme is higher than an instance with low performance and is lower than an instance with high performance.

H.-J. Jeong et al. propose a multilayered quad rotor control method that can move the quad rotor to the desired goal while resisting disturbance. The proposed control system is modular, convenient to design and verify, and easy to extend. It comprises three layers: a physical layer, a displacement control layer, and an attitude control layer. The displacement control layer considers the movement of the vehicle, while the attitude control layer controls its attitude. The physical layer deals with the physical operation of the vehicle. The two control layers use a mathematical method to provide minute step-by-step control. The proposed control system effectively combines the three layers to achieve drift stabilization.

K.-S. Wong and M. H. Kim studied a self-awareness data collection protocol to raise the confidence of the respondents when submitting their personal data to the data collector. Their self-awareness protocol requires each respondent to help others in preserving individual privacy. The communication (respondents and data collector) and collaboration (among respondents) in their solution will be performed automatically.

H. K. Lee et al. provided the following contributions; they presented a service design for an adaptive STB that decreases the dependence among scalable layers. Their adaptive STB converts the receiving scalable streams with high dependency into scalable streams with low dependency. As a result, it decreases the indirect loss of media data and increases streaming service performance even over mobile networks. They then analyzed a media scheme to convert scalable streams.

S. Sivamani et al. proposed a smart logistics service model for providing user-centric intelligent logistics service by utilizing smartphones in a smart environment. They also develop an OWL based ontology model for the smart logistics for the better understanding among the context information.

S.-H. Chae et al. researched how to reconstruct segmentation region in a small region in order to improve the segmentation results. They generated predicted segmentation of slices using volume data with linear equation and proposed improvement method for small regions using the predicted segmentation. In order to verify the performance of the proposed method, lung region by chest CT images was segmented. As a result of experiments, volume data segmentation accuracy rose from 0.978 to 0.981 and from 0.281 to 0.187 with a standard deviation improvement confirmed.

J.-H. Park et al. provided a higher level of abstraction by integration of the two types of access interface, REST API and MapReduce. The motivation of this research stems from the slower response time for accessing simple RDBMS on Hadoop than direct access to RDBMS. This is because there is overhead to job scheduling, initiating, starting, tracking, and management during MapReduce-based parallel execution. Therefore, they provided a good performance for REST Open API service and for MapReduce, respectively.

D. Jung et al. proposed an estimated interval-based checkpointing (EIC) using weighted moving average. Their scheme sets the thresholds of price and execution time based on history. Whenever the actual price and the execution time cross over the thresholds, the system saves the state of spot instances. The Bollinger Bands is adopted to inform the ranges of estimated cost and execution time for user's discretion. The simulation results reveal that, compared to the HBC and REC, the EIC reduces the number of checkpoints and the rollback time. Consequently, the task execution time has been decreased with EIC by HBC and REC. The EIC also provided the benefit of the cost reduction by HBC and REC, on average.

D. Kang et al. discussed their approach and confirm the validity of their proposed system for preventing network and application protocol attacks in SCADA sensor networks.

J. Song et al. presented a novel approach called TSMC (three-stage memory copy) for live virtual machine migration. In TSMC, memory pages only need to be transmitted twice at most and page fault just occurred in small part of dirty pages. They implement it in Xen and compare it with Xen's original precopy approach. The experimental results under various memory workloads show that TSMC approach can significantly reduce the cumulative migration time and total pages transferred and achieve better network IO performance at the same time.

S.-Y. Ihm et al. propose a new skyline method (called Grid-PPPS) for efficiently handling top- k queries in IoT applications. The proposed method first performs Grid-based partitioning on data space and then partitions it once again using hyperplane projection. Experimental results show that their method improves the index building time compared to the existing state-of-the-art methods.

K. Hwang and S. Nam introduced a DSME beacon scheduling model and present a concrete design model.

Furthermore, validity and performance of DSME are evaluated through experiments. Based on experiment results, they analyzed the problems and limitations of DSME, presented solutions step by step, and finally proposed an enhanced DSME beacon scheduling model. Through additional experiments, they proved the performance superiority of enhanced DSME.

R. Su et al. compared the PMPGA (parallel multipopulation genetic algorithm) with the multiobjective fuzzy optimization algorithm and differential evolution based algorithm and showed that PMPGA has achieved better result. The method can be widely applied to related high-speed train.

K.-C. Lin et al. proposed a classified model in which an artificial fish swarm algorithm and a support vector machine are combined. ALAN environment with several computers which has been infected by the botnet virus was simulated for testing this model; the packet data of network flow was also collected. The proposed method was used to identify the critical features that determine the pattern of botnet. The experimental results indicated that the method can be used for identifying the essential botnet features and that the performance of the proposed method was superior to that of genetic algorithms.

K.-C. Chu and L.-P. Hung proposed the TRPN model and provided a practical, effective, and adaptive method for risk evaluation. In particular, the defined GRPN function offers a new method to prioritize failure modes in failure mode and effect analysis (FMEA). The different risk preferences considered in the healthcare example show that the modified FMEA model can take into account the various risk factors and prioritize failure modes more accurately.

S.-H. Park et al. proposed energy-efficient probabilistic routing (EEPR) algorithm, which controls the transmission of the routing request packets stochastically in order to increase the network lifetime and decrease the packet loss under the flooding algorithm. The proposed EEPR algorithm adopts energy-efficient probabilistic control by simultaneously using the residual energy of each node and ETX metric in the context of the typical AODV protocol. In the simulations, they verified that the proposed algorithm has longer network lifetime and consumes the residual energy of each node more evenly when compared with the typical AODV protocol.

To overcome this limitation, H. Cho et al. tested the usability of a new conceptual autonomous emergency braking (AEB) system that employs vehicle-to-vehicle (V2V) communication technology in the existing AEB system. To this end, a radar sensor and a driving and communication environment constituting the AEB system were simulated; the simulation was then linked by applying vehicle dynamics and control logic. The simulation results show that the collision avoidance relaxation rate of V2V communication-based AEB system was reduced compared with that of existing vehicle-mounted-sensor-based system. Thus, a method that can lower the collision risk of the existing AEB system, which uses only a sensor cluster installed on the vehicle, is realized.

Y.-S. Jeong et al. proposed an efficient and secure mobile-IPS (m-IPS) for businesses utilizing mobile devices in mobile environments for human-centric computing. The m-IPS

system incorporates temporal spatial awareness in human-centric computing with various mobile devices and checks users' temporal spatial information, profiles, and role information to provide precise access control. And it also can extend application of m-IPS to the Internet of Things (IoT), which is one of the important advanced technologies for supporting human-centric computing environment completely, for real ubiquitous field with mobile devices.

L. Chen et al. studied the exponential stability of the complex dynamical network described by differentially nonlinear equations which couple with time delay and stochastic impulses. Some sufficient conditions are established to ensure p th moment exponential stable for the stochastic impulsive systems (SIS) with time delay. An example with its numerical simulation is presented to illustrate the validation of main results.

Finally, D. Shin et al. developed watch-type device (smart watch) that patients wear and a server system. The smart watch developed includes a GPS, accelerometer, and illumination sensor and can obtain real-time health information by measuring the position of patients, quantity of exercise, and amount of sunlight. The server system includes the sensor data analysis algorithm and web server used by the doctor and protector to monitor the sensor data acquired from the smart watch. The proposed data analysis algorithm acquires the exercise information and detects the step count in patients' motion acquired from the acceleration sensor and verifies the three cases of fast pace, slow pace, and walking pace, showing 96% of the experimental results.

Acknowledgments

Our special thanks go to everybody of Journal of Applied Mathematics. We would like to thank all authors for their contributions to this special issue. We also extend our thanks to the external reviewers for their excellent help in reviewing the manuscripts.

Young-Sik Jeong
 Mohammad S. Obaidat
 Jianhua Ma
 Laurence T. Yang

References

- [1] Y.-S. Jeong, N. Chilamkurti, and L. J. G. Villalba, "Advanced technologies and communication solutions for internet of things," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 896760, 3 pages, 2014.
- [2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [3] Y.-S. Jeong and J. H. Park, "High availability and efficient energy consumption for cloud computing service with grid infrastructure," *Computers and Electrical Engineering*, vol. 39, no. 1, pp. 15–23, 2013.
- [4] Gartner, *Gartner's Hype Cycle Special Report for 2011*, Gartner, 2012, <http://www.gartner.com/technology/research/hype-cycles/>.

- [5] H. Ning and Z. Wang, "Future internet of things architecture: like mankind neural system or social organization framework?" *IEEE Communications Letters*, vol. 15, no. 4, pp. 461–463, 2011.

Research Article

Investigation Methodology of a Virtual Desktop Infrastructure for IoT

Doowon Jeong,¹ Jungheum Park,¹ Sangjin Lee,¹ and Chulhoon Kang²

¹Center for Information Security Technologies (CIST), Korea University, Anam-dong, Seongbuk-gu, Seoul 136-713, Republic of Korea

²Supreme Prosecutors' Office, Seocho-dong, Seocho-gu, Seoul 137-730, Republic of Korea

Correspondence should be addressed to Sangjin Lee; sangjin@korea.ac.kr

Received 13 March 2014; Accepted 31 July 2014

Academic Editor: Young-Sik Jeong

Copyright © 2015 Doowon Jeong 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.

Cloud computing for IoT (Internet of Things) has exhibited the greatest growth in the IT market in the recent past and this trend is expected to continue. Many companies are adopting a virtual desktop infrastructure (VDI) for private cloud computing to reduce costs and enhance the efficiency of their servers. As a VDI is widely used, threats of cyber terror and invasion are also increasing. To minimize the damage, response procedure for cyber intrusion on a VDI should be systematized. Therefore, we propose an investigation methodology for VDI solutions in this paper. Here we focus on a virtual desktop infrastructure and introduce various desktop virtualization solutions that are widely used, such as VMware, Citrix, and Microsoft. In addition, we verify the integrity of the data acquired in order that the result of our proposed methodology is acceptable as evidence in a court of law. During the experiment, we observed an error: one of the commonly used digital forensic tools failed to mount a dynamically allocated virtual disk properly.

1. Introduction

In the recent past, cloud computing has experienced phenomenal growth for IoT (Internet of Things). To offer IoT services, many companies have managed to reduce costs and enhance the efficiency of their servers by adopting a virtual desktop infrastructure (VDI) which is classified into private cloud computing. Private cloud computing involves the use of virtualization technology of cloud servers. Resources such as CPU, RAM, and server storage are shared. Unlike a public cloud, the servers are only used by internal users. The use of private cloud computing is continually increasing owing to its efficiency.

However, as VDI is widely used, threats of cyber terror and invasion are also increasing. In VDI, all resources are shared; it would be critical to whole users. To minimize the damage, response procedure such as investigating causal relationship and identifying a criminal on a VDI should be systematized either scientifically or technically. However, investigation methodology for private clouds are not keeping

pace with this growth in private cloud computing, despite much research into investigation and digital forensics for cloud computing. Taylor et al. outlined challenges and considerations relevant to examiners when investigating general cloud computing environments [1]. Chung et al. proposed a procedure for investigating and analyzing artifacts for users of cloud storage services [2]. Dykstra and Sherman researched a forensic collection method for infrastructure-as-a-service cloud computing [3]. However, to the best of our knowledge, research on digital forensic investigation (DFI) for a complete VDI has yet to be accomplished. Other research into digital forensics for cloud computing tends to focus on concepts or processes for general investigation and evidence collection. Therefore, more research into DFI for VDI is necessary.

In cloud-hosted virtual desktop environments, user data may not be stored on the local system but in distributed storage linked by a hypervisor, unlike noncloud-hosted virtual desktop environments [4–6]. An investigation of a computer requires an image of the entire target device [7]. However, this is becoming increasingly impractical because storage

TABLE 1: Hypervisor and desktop virtualization solutions.

Component	Citrix	VMware	Microsoft
Hypervisor	XenServer 6.0	ESXi Server 5.0	Hyper-V (Windows Server 2008 R2)
Hypervisor management system	XenCenter 5.6	View 5.0	Hyper-V (Windows Server 2008 R2)

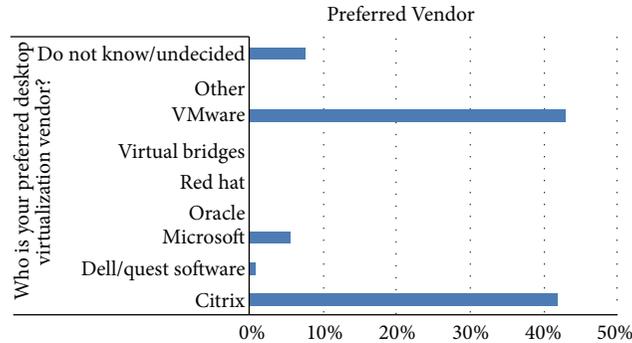


FIGURE 1: Desktop virtualization trends at the Gartner Data Center [12].

can contain many terabytes of data. Partial or selective file copying such as a virtual hard disk for a specific user may be considered for DFI in a cloud computing environment [8–11]. Therefore, we believe that this new approach will be very useful for investigating crimes and causal relationship related to VDI invasion accident.

The remainder of the paper is organized as follows. In Section 2, we present VDI for IoT and briefly introduce popular desktop virtualization solutions, such as VMware, Citrix, and Microsoft. In Section 3, we propose a DFI method that searches for user traces, assigns information between a user and a virtual machine, and collects data using structural features and functions of each desktop virtualization solution. In Section 4, we verify the integrity of VDI data acquisition in an experiment. In Section 5, we report an error identified during this experiment: Encase, a widely known digital forensic tool, failed to mount a dynamically allocated virtual disk properly. Section 6 concludes.

2. Virtual Desktop Infrastructure

2.1. Desktop Virtualization Solutions. In computing, virtualization is a technique for sharing resources such as hardware platforms, operating systems, storage, and network devices [13, 14]. Desktop virtualization involves separating the logical desktop from the physical server, which is realized by a hypervisor. A hypervisor is a logical platform for simultaneous operation of multiple operating systems on a host server. VDI is a desktop-centered service that hosts user desktop environments on remote servers and/or blade PCs; the hosts can access VDI over a network using a remote display protocol. Desktop virtualization solutions are software packages consisting of several programs, and these solutions are based on the hypervisor. There are various desktop virtualization solutions; Citrix, VMware, and Microsoft are the most popular (Figure 1). Therefore, we focused on these three solutions here. Each solution has its own hypervisor: Citrix uses

XenServer, VMware uses ESX/ESXi Server, and Microsoft uses Hyper-V. Here, we construct a VDI that consists of a hypervisor and a desktop virtualization solution. Table 1 lists the hypervisor versions and desktop virtualization solutions we used in the study.

2.2. VDI Structure. Although the hypervisor and desktop virtualization solution comprising each VDI differ, a survey revealed that the configuration methods are very similar [15–17] (Table 2). As shown in Figure 2, a hypervisor and hypervisor management system are required to create and manage virtual machines. A local storage device such as the hard disk of a hypervisor system can be used as a storage unit for the virtual machine. However, in the majority of cases, shared storage devices are used because companies require many virtual machines to offer private cloud computing services to their members. An authentication management system and a connection management system are also essential for user authentication and delivery of a virtual machine to the user. A user can access the virtual machine using a specific program or web once the VDI is constructed. The access process for the virtual machine is as follows (Figure 2).

- (1) A connect request (login) is sent to the connection management system.
- (2) The connection management system sends the user login information to the authentication management system.
- (3) On successful user authentication, the connection management system asks the hypervisor to assign a virtual machine, which is stored in the shared storage.
- (4) The connection management system delivers that virtual machine to the user.
- (5) Then, the virtual machine can be used as a personal desktop.

TABLE 2: VDI components.

Component	Citrix	VMware	Microsoft	Role
Hypervisor	XenServer	ESXi server	Hyper-V	Create and manage virtual machines
Hypervisor management system	XenCenter	vCenter server	SCVMM (system center virtual machine manager)	Manage the hypervisor
Connection management system	DDC (desktop delivery controller)	View Manager	RDCM (remote desktop connection manager)	Connect and assign a virtual machine to a user
Authentication management system	Active Directory	Active Directory	Active Directory	Register (create/delete) and authenticate the user
Virtual machine access program	Web browser (Citrix receiver should be installed)	View client or web browser	Web browser	Access to virtual machine

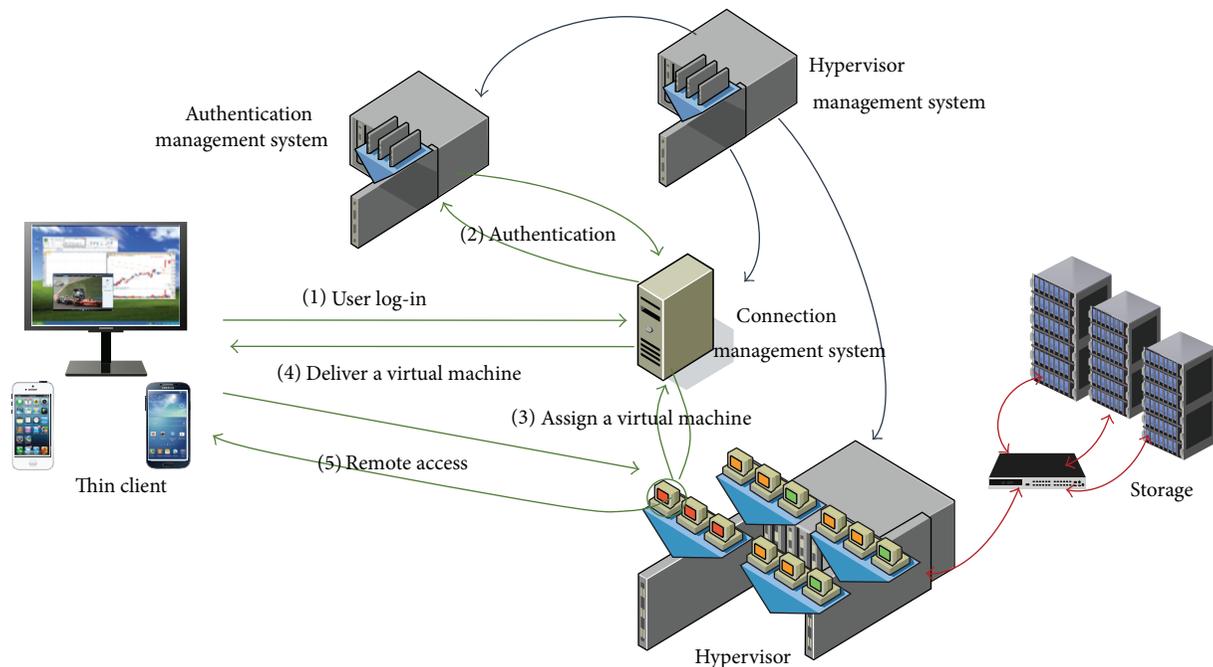


FIGURE 2: General VDI structure.

3. DFI Method for VDI

In VDI, user data are stored in the central storage for virtual machines. There are two methods for gathering a user's data: one is to investigate the entire central storage, and the other is to remotely extract the virtual machine allocated to that user. The first method is inefficient because the central storage capacity is huge and so investigation is very time consuming. Therefore, the second method is preferable because it is similar to disk imaging for investigation of the hard disk of a local desktop. Hence, extraction of a virtual machine is the main point for investigating a VDI. To achieve this, an investigator must determine whether or not the suspect uses a particular virtual machine.

DFI for VDI targets systems that carry user traces. The trace recorded by a system is used to access the virtual machine. To find the trace, the first step is to investigate

the thin client for a user using the virtual desktop as in Figure 3. When a user accesses a virtual machine, access information such as registry data, log files, or web history is recorded in the thin client and can be discovered via a signature search, depending on the solution. However, if this information cannot be uncovered (e.g., the records have been deleted and the programs have been removed), it is difficult to obtain virtual machine access information from the thin client. In this case, the investigator only needs to check the user access information and virtual machine assignment information in the connection management system and the authentication management system.

After inspecting the relevant virtual machine access information, the investigator should collect data for the virtual machine used by the suspect. For this, the investigator requires administrator authority for the hypervisor

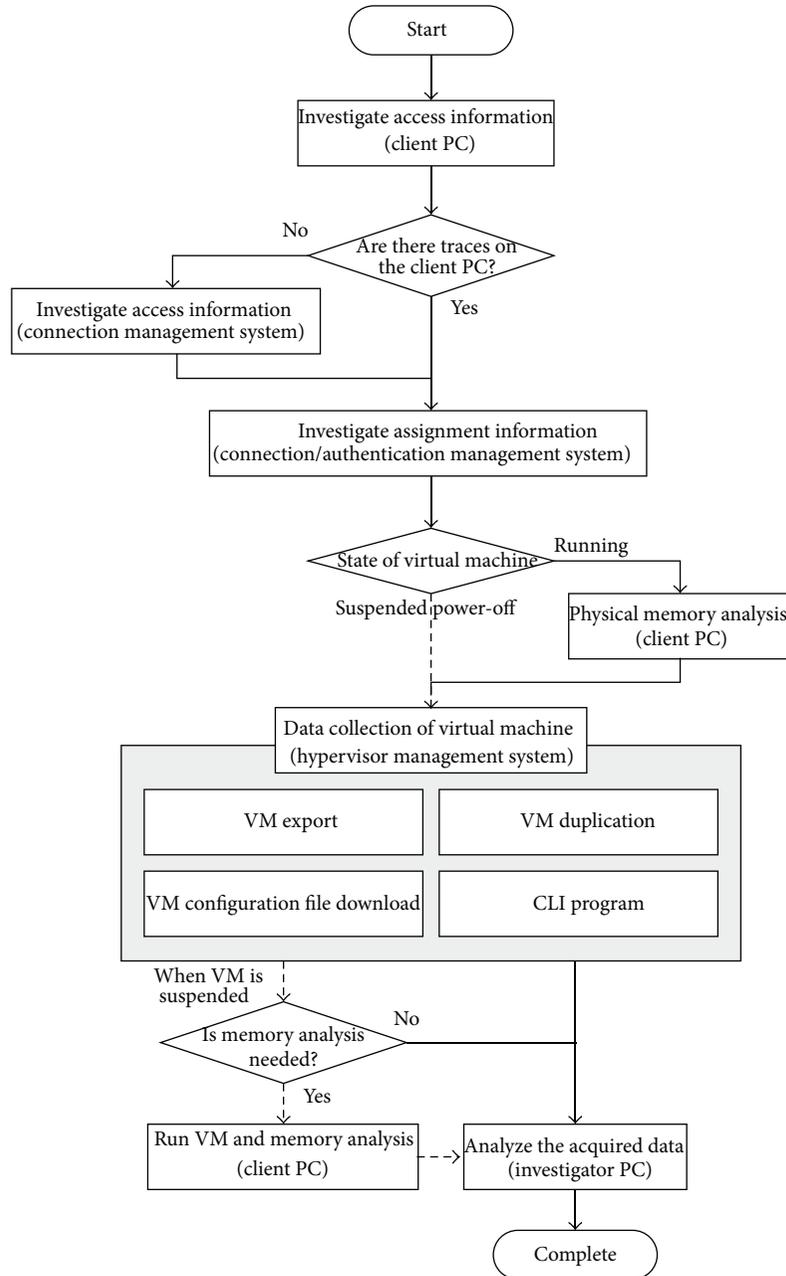


FIGURE 3: Digital forensics procedure for VDI in private cloud computing.

or its management system or user authority for the virtual machine. If access authorities are obtained, then the data can be collected via the hypervisor management system, shell connection, or virtual machine access. Data collection via the hypervisor management system or shell connection requires a dedicated program for each solution. If the virtual machine is already running, the investigator can analyze live memory and perform a memory dump by executing memory forensics tools in the virtual machine. Detailed information is presented in Section 3.3. The collected data can then be analyzed using general DFI methods and tools.

Here, we make two assumptions: (i) the investigator already knows the suspects, because private cloud computing

services are provided to restricted users who have access authority; and (ii) the investigator has administrator or user authority with assistance from the organization.

3.1. User Access Information. As mentioned above, the VDI structure of Citrix, VMware, and Microsoft is very similar. Therefore, the DFI method is similar to these solutions. Evidence of use of a virtual machine is logged in the user's computer, hypervisor management system, connection management system, and authentication management system. Here, a DFI method for a general VDI using Citrix, VMware, and Microsoft and local computers operating on Windows 7, Ubuntu 12.04, and Mac OS 10.8.2 is studied.

TABLE 3: Access information for a virtual machine logged in the local Windows system.

Solution	Registry	Log/web browser signature
Citrix	<i>KEY_CURRENT_USER\Software\Citrix\XenDesktop\DesktopViewer\[/VM name]</i> ⇒ VM name, IP address of connection management system (DDC)	<i>%UserProfile%\AppData\Roaming\ICAClient</i> ⇒ VM name, connection/disconnection time
		Signature: DesktopWeb ⇒ connection time, IP address or name of connection management system (DDC)
VMware	<i>HKEY_CURRENT_USER\Software\VMware, Inc.\VMware VDM\Client</i> ⇒ VM name, IP address or URL of connection management system (View Manager), domain name, user computer name	<i>%UserProfile%\AppData\Local\VMware\VDM\logs</i> ⇒ URL of connection management system (View Manager), connection/disconnection time, domain name, user computer name
		※ log-[yyyy]-[mm]-[dd].txt
Microsoft	<i>KEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Default</i> ⇒ VM name or IP address	Signature: RDWeb ⇒ connection time, Hyper-V server name, domain name

3.1.1. *Traces on the Client PC.* In Windows 7, registry and log entries are created when VMware is used. When Citrix is used, registry, log entries, and web history traces are created. When a Microsoft VDI is used, registry entries related to the remote desktop are created, but log entries are not created. However, Microsoft uses a specific signature, RDWeb, when a connection using the web is made to a virtual desktop environment. Therefore, access information can be determined from the web history. Table 3 shows the access information for a virtual desktop environment logged in the local Windows system.

In Ubuntu 12.04 and Mac OS 10.8.2, access information for VMware can be found from the log created as in Windows OS. However, for Citrix, when a connection is made via a web browser, an investigator should check the history of the web browser. Thus, we studied Firefox, the default web browser in Ubuntu, and Safari, the default web browser in Mac OS. Further, for Microsoft, unlike in Windows, access information cannot be found via web history analysis since an RDWeb connection is impossible using web browser in Ubuntu and Mac OS. Instead, the access information can be found from the information retained when a remote desktop connection from each OS to the Microsoft virtual machine is made. Table 4 shows the access information logged in local Ubuntu and Mac systems.

3.1.2. *Traces on the Connection Management System.* If there are no connection traces on the user's local computer, the investigator should focus on the connection management system, which assigns virtual machines to users, manages the machines, and connects or disconnects virtual machines according to user requests. Therefore, all information pertaining to connections to virtual machines is managed and logged here. An investigator can find information on the exact time at which a user connected to or disconnected from the virtual machine by analyzing these log files. Table 5 shows the access information logged in the connection management system.

3.2. *Virtual Machine Assignment Information.* To connect to a virtual machine, a user must be assigned a virtual machine through the connection management system. A virtual machine assigned to a specific user cannot be accessed by others and will be used only by that user. The assignment information is stored in the connection management system and authentication management system. It is useful to prove the relationship between a suspect and a virtual machine. The assignment information in the connection management system should be investigated to establish connection information between the virtual machine and its user. Table 6 shows how to find this assignment information in the connection management system. Assignment information is also stored in the database of the connection management system or authentication management system. Table 7 summarizes the method for finding assignment information between a user and a virtual machine from the database.

3.3. *Data Collection for a Virtual Machine.* In a virtual desktop environment, data for a virtual machine are stored in the storage area for the server and not on the local computer. Therefore, an investigator should investigate the central storage area. However, when a cloud environment is constructed, the central storage area is typically made up of multiple independent storage devices [18]. It is not feasible to collect all the data from these devices. Thus, it is most efficient to acquire a virtual hard disk for the virtual machine. However, it is difficult to acquire data for a virtual machine because the virtual hard disk can be allocated in various ways: as single or multiple files and via static or dynamic allocation. The data could be stored on one physical disk or distributed over multiple disks. Therefore, we use the hypervisor management system and shell connection program for each solution to acquire a virtual hard disk for the suspect because data extraction is possible without reference to the type of allocation. If a user is connected to a virtual machine, the investigator can collect data such as a memory dump, specific files, or the entire virtual hard disk of the virtual machine.

TABLE 4: Access information logged in local Ubuntu and Mac systems.

Solution	Ubuntu 12.04	Mac OS 10.8.2
Citrix	Cache: <code>\home\[user name]\.mozilla\firefox\6lhwv183.default\Cache_CACHE_[numbers]_</code> History: <code>\home\[user name]\.mozilla\firefox\6lhwv183.default\places.sqlite</code> Cookie: <code>\home\[user name]\.mozilla\firefox\6lhwv183.default\cookies.sqlite</code> Session: <code>\home\[user name]\.mozilla\firefox\6lhwv183.default\sessionstore.js</code> ⇒ IP address or URL of connection management system (DDC)	Cache: <code>\Users\[user name]\Library\Caches\com.apple.Safari\Cache.db</code> History: <code>\Users\[user name]\Library\Safari\History.plist</code> Cookie: <code>\Users\[user name]\Library\Safari\Cookies.plist</code> Session: <code>\Users\[user name]\Library\Safari>LastSession.plist</code> ⇒ IP address or URL of connection management system (DDC)
	<code>\tmp\vmware-[user name]\vmware-view-[numbers].logs</code> ⇒ IP address or URL of connection management system (View Manager), connection/disconnection time, user ID, VM name, domain name	<code>\Users\[user name]\Library\Logs\VMware View Client\vmware-view.logs</code> ⇒ IP address or URL of connection management system (View Manager), connection/disconnection time, VM IP address, domain name
Microsoft	<code>\home\[user name]\.bash_history</code> ⇒ VM name or IP address, user ID (option), user password (option), domain name (option)	<code>\Users\[user name]\Documents\RDC Connections\Default.rdp</code> ⇒ VM name, user ID, domain name

TABLE 5: Access information logged in the connection management system.

Solution	Log
Citrix	<code>%SystemDrive%\inetpub\logs\LogFiles\[folder name]</code> ⇒ connection/disconnection time, connection management system (DDC) and user IP address × <code>[yymmdd].log</code>
VMware	<code>%SystemDrive%\ProgramData\VMware\VDM\logs</code> ⇒ VM name and IP address, connection/disconnection/reconnection/logoff time, domain name, user computer name × <code>log-[yyyy]-[mm]-[dd].txt</code>
Microsoft	<code>%SystemDrive%\inetpub\logs\LogFiles</code> ⇒ connection/disconnection time, user ID × <code>[yymmdd].log</code>

TABLE 6: Method for finding assignment information in the connection management system.

Solution	Method
Citrix	DDC (1) Start Citrix Desktop Studio on DDC (2) Select Desktop Studio-Assignments (3) Select VM or Group
	View Manager (1) Start View Administrator Console on View Manager (2) Select Inventory-Desktops
	Active Directory (1) Start Active Directory user and computer on Active Directory (2) Select user-properties—personnel virtual desktop

3.3.1. *Hypervisor Management System.* A target virtual machine can be exported or duplicated and the component files can be downloaded using the hypervisor management

system provided by each solution. Table 8 summarizes methods for collecting virtual machine data using the hypervisor management system.

When using VM export, the virtual machine data are converted to the solution format (e.g., xva file format for Citrix). VM duplication means that the raw data for the virtual machine can be obtained. In the case of VMware, we can select and download some configuration files using the VM configuration file download method.

3.3.2. *Shell Connection Program.* Each solution provides a command-line interface (CLI) with various administrative and management-oriented utilities. One such utility provided by each solution allows acquisition of a copy of the state of the virtual machine. VMware and Microsoft can collect the raw data duplicated from the original virtual disk. Citrix, however, can only collect compressed data. Thus, XenCenter is required to recover and analyze virtual machine data hosted and acquired via Citrix. Table 9 summarizes the method for

TABLE 7: Method of finding assignment information in the database of the connection or authentication management system.

Solution	Method
Citrix	DDC (1) Connect to DB by using MS SQL Server Management Studio (2) [DDC PC name]-[Databases]-[CitrixXenDesktopDB]-[Tables]-[chb_Stat e.AccountNames]: user name and Uid (3) [DDC PC name]-[Databases]-[CitrixXenDesktopDB]-[Tables]-[chb_Stat e.WorkerDiags]: VM assigned user (Uid)
	View Manger (ADAM DB) (1) Connect to ADAM DB by using Active Directory Explorer (2) [DC=vdi,DC=vmware,DC=int]-[OU=Servers]: specific VM CN (Common Name) value and other information (a) Description: VM name (b) Member: user CN (3) [DC=vdi,DC=vmware,DC=int]-[CN=ForeignSecurityPrinciple]: user CN value and other information (a) Description: user and domain name
	Active Directory (ADAM DB) (1) Connect to ADAM DB by using Active Directory Explorer (2) [DC=domain name]-[OU=Hyper-V]: user name and other information (a) msTSPrimaryDesktop: assigned VM name

TABLE 8: Data acquisition method using the hypervisor management system.

Solution	VM export	VM duplication	VM configuration file download
Citrix (XenCenter)	Select VM-Menu-VM-Export ⇒.xva or.ovf file Export	Select VM-click mouse right button-Copy VM-Full copy	
VMware (vCenter)	Select VM-Menu-File- Export-OVF Template Export ⇒.ovf file Export	Select VM-duplication	Select Hypervisor or VM-Summary-Resource- Storage-select Datastore-Browse Datastore-select folder or file-download
Microsoft (Hyper-V Manager and SCVMM)	Hyper-V Manager-select VM-click mouse right button-Export ⇒.vhd file Export	SCVMM-select VM-duplication-deploy VM on host	

collecting virtual machine data using the shell connection program.

3.3.3. Consideration of the State of a Virtual Machine. In a virtual desktop environment, a virtual machine can be running, suspended, or in a power-off state. An investigator should check the state of a virtual machine before acquiring data, because the acquisition method that is applicable varies, depending on the state. Table 10 lists applicable acquisition methods. It is evident that when the virtual machine is running, it is impossible to acquire the virtual disk using the Citrix and Microsoft solutions. For the Microsoft solution, the investigator should turn off the virtual machine. If analysis of the memory is essential, the investigator should analyze the memory before turning off or suspending the virtual machine. For analysis of the memory when the virtual machine is in a suspended state, the investigator should first acquire the virtual disk and then resume the virtual machine for memory forensics.

4. Verification of Acquisition Data Integrity

The integrity of the acquired data should be demonstrated for admissibility of evidence in a court of law. Hence, in this

section, we verify the integrity of the virtual hard disk drive (HDD) acquired according to our method.

4.1. Experiment #1: Comparison of Hash Values for the Original Virtual HDD and the Acquisition Data. Several methods can be used to acquire a virtual hard disk. In VMware, acquisition is via a shell connection program and VM export, duplication, and file download through the hypervisor management system. As Microsoft and Citrix do not provide VM file download, we acquire data via a shell connection program and VM export or duplication through the hypervisor management system. After acquiring the data, we compared hash values for the original virtual HDD of the virtual machine and the acquisition data. Table 11 lists the results.

For VMware and Microsoft, the hash values match perfectly, regardless of the acquisition method used. The sizes of the original virtual HDD and acquisition data are also the same. Therefore, investigation using VMware or Microsoft according to the proposed acquisition method yields that data are admissible as evidence in a court of law.

However, for Citrix, the hash values are different. First, there is a difference between the format of the original virtual HDD data and the acquisition data. The format of the original

TABLE 9: Acquisition of virtual machine data using a hypervisor CLI with default utilities for each solution.

Solution	Shell connection program
Citrix (XenCenter console Tab)	Connect to shell or select “Console” tab on XenCenter Virtual disk collection: xe vm-export vm=[VM name] filename=[file mane].xva
VMware (vSphere PowerCLI)	Connect to shell using vSphere PowerCLI Virtual disk collection command: copy-datastoreitem [datastore drive]:\[Src. path] [Dst. path] *vSphere PowerCLI should be installed
Microsoft (Windows PowerShell)	Connect to shell using Windows PowerShell Virtual disk collection command: export-vm-vm “[VM name]”-server [Hyper-V Server name]-path [Dst. path] *PowerShell Management Library for Hyper-V should be installed

TABLE 10: Applicable acquisition method depending on the solution and state of the virtual machine.

Solution	Acquisition method	State		
		Running	Suspended	Power-off
Citrix	VM export	No	Yes	Yes
	VM duplication	No	Yes	Yes
	VM configuration file download	No	No	No
	CLI program	No	Yes	Yes
VMware	VM export	No	No	Yes
	VM duplication	Yes	Yes	Yes
	VM configuration file download	No	Yes	Yes
	CLI program	No	Yes	Yes
Microsoft	VM Export	No	No	Yes
	VM duplication	No	No	Yes
	VM configuration file download	No	No	No
	CLI program	No	No	Yes

TABLE 11: Results for experiment #1 on integrity verification.

Solution	Acquisition method	Hash value		Result
		Original virtual HDD	Acquisition data	
VMware	VM export		0440B1A068A0A9D116B2184E824196D7	Match
	VM duplication	0440B1A068A0A9D116B2184E824196D7	0440B1A068A0A9D116B2184E824196D7	Match
	VM file download		0440B1A068A0A9D116B2184E824196D7	Match
	CLI program		0440B1A068A0A9D116B2184E824196D7	Match
Citrix	VM export		06D6A00AD0A51EFE1E31B04B0D473BE2 (Disk size: 5,200,160,256 bytes)	Mismatch
	VM duplication	CEDB64BD9510566BD3A7A516CADF6444 (Disk size: 5,309,903,360 bytes)	06D6A00AD0A51EFE1E31B04B0D473BE2 (Disk size: 5,200,160,256 bytes)	Mismatch
	CLI program		06D6A00AD0A51EFE1E31B04B0D473BE2 (Disk size: 5,200,160,256 bytes)	Mismatch
Microsoft	VM export		328D07681CD90C98BB71F625F47B3F07	Match
	VM duplication	328D07681CD90C98BB71F625F47B3F07	328D07681CD90C98BB71F625F47B3F07	Match
	CLI program		328D07681CD90C98BB71F625F47B3F07	Match

data is VHD, but that of the acquisition data is XVA or OVF and the data are compressed. Decompression of an acquisition file leads to a smaller size than of the original. This is because Citrix rearranges the original data when the data are acquired via XenCenter. Figure 4 shows that the offset of a specific file is changed from 0x10CFFF to 0x10C800.

Repetition of the experiment revealed that when data are acquired or duplicated using XenCenter, they are transmitted via blocks and the transmitted data are rearranged. It is impossible to verify the integrity of the original virtual HDD and the acquisition data by comparing hash values because the data order is inverted when the original HDD is acquired.

TABLE 12: Results for experiment #2 on integrity verification for logical drives for Citrix.

Area	Original HDD	Acquisition data	Result
Boot	092D9487556456C6881F16BEA9FABCD A	092D9487556456C6881F16BEA9FABCD A	Match
Data	27A83C3709DEE6F042AA064C56B7DE29	27A83C3709DEE6F042AA064C56B7DE29	Match

10:CFF0h:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 EBè
10:D000h:	52 90 4E 54	46 53 20 20	20 20 00 02	08 00 00 00		R.NTFS.....
10:D010h:	00 00 00 00	F8 00 00 3F	00 FF 00 00	08 00 00 00		...ø...?..ÿ.....
10:D020h:	00 00 00 80	00 80 00 FF	1F 03 00 00	00 00 00 55		...€..€..ÿ.....U
10:D030h:	21 00 00 00	00 00 00 02	00 00 00 00	00 00 00 F6		!.....ö
10:C7F0h:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	
10:C800h:	EB 52 90 4E	54 46 53 20	20 20 00 02	08 00 00 00		ëR.NTFS.....
10:C810h:	00 00 00 00	00 F8 00 00	3F 00 FF 00	00 08 00 00	ø...?..ÿ.....
10:C820h:	00 00 00 00	80 00 80 00	FF 1F 03 00	00 00 00 00		...€..€..ÿ.....
10:C830h:	55 21 00 00	00 00 00 00	02 00 00 00	00 00 00 00		U!.....

FIGURE 4: Comparison of offsets for the same file: top, original HDD; bottom, acquisition data.

4.2. Experiment #2: Comparison of Hash Values for Logical Drives. The integrity of Citrix acquisition data was verified in a different manner. We mounted the original HDD and the acquisition data on a local computer to verify the integrity. The hash value for each logical drive was then calculated. Various tools were used to enhance the reliability of the experimental results. The tools Mount Image Pro, FTK Imager, and X-Way Forensics were used for mounting the disk image, and Encase, FTK Imager, and X-Way Forensics were used for calculating hash values. The reason why Encase was not used for mounting is explained in Section 5. Table 12 lists the results for these experiments.

Table 12 reveals that the size and hash values match for the original HDD and the acquisition data. We also verified the integrity of the acquisition data by comparison of hash values for each mounted logical drive. The results for experiments #1 and #2 prove that the proposed acquisition method ensures data integrity.

5. Reliability Verification of Forensic Tools for Virtual Machine Data

During experiment #2, we found that Encase 6 and Encase 7 could not parse the acquired data in their entirety when mounting the virtual HDD, which is dynamically allocated in VHD format. This problem was observed both for data acquired through Citrix and for the Microsoft solution. To explore this problem further, we compared various tools. Table 13 shows the ability of each tool to correctly parse the acquired dynamic VHD formats.

Encase failed to properly mount the original virtual HDD as well as the copy. To understand the reason behind this problem, we calculated the hash values for all the entries for virtual drives mounted by Encase, FTK Imager, and X-Way Forensics. There were 59,127 entries and the hash values for 13 of these entries were mismatched.

To analyze this issue in detail, we compared the mismatched files using a hex editor. As observed in Figure 5, the hex values are different even though they are at the same offset in the same file (pagefile.sys). We found that unknown values were repeatedly written at a specific offset for some files, but the reason why these are written when Encase mounts a dynamic VHD format remains unknown.

This finding indicates that an investigator should avoid Encase when mounting acquired data in a dynamic VHD format. However, Encase may be used to analyze the data after mounting via some other tool.

6. Conclusion

Adoption of a VDI for IoT can save costs and is a convenient alternative for users. However, investigation methods for VDI invasion accidents have not kept pace with the VDI market, which is rapidly growing and experiencing wide development.

Here, we explained VDI and popular VMware, Citrix, and Microsoft desktop virtualization solutions. The infrastructure of the three solutions is very similar, so we were able to establish a framework for VDI investigation. Since VDI is different from general PC environments, we focused on acquiring the data for a virtual machine using user access information from the PC thin client, the connection management system, and the authentication management system. By applying the proposed method to VDI, an investigator can obtain a virtual disk image and analyze this as for general disk forensics. We verified the integrity of data acquired via our method through experiments for admissibility of evidence in a court of law. Moreover, we discovered that a widely used tool has an error and failed to properly mount acquired data in a dynamic VHD format.

This paper will be useful for investigation of cases in which VDI plays an essential role. We hope that it will inspire further research on DFI methods in response to the rapidly growing cloud computing environment.

TABLE 13: Comparison of hash values for various tools.

Index	Original virtual hard disk	Copy of virtual hard disk	Result
EnCase	C69289228xxxxxx	64C4D1298xxxxxx	Mismatch
FTK	C5F64F49Cxxxxxx	C5F64F49Cxxxxxx	Match
X-Way Forensics	C5F64F49Cxxxxxx	C5F64F49Cxxxxxx	Match

14:5FD0h:	6E 6B 20 00	20 E4 27 92	6C 7E CD 01	00 00 00 00	nk . ä' 1~ Í
14:5FE0h:	E0 22 00 00	00 00 00 00	00 00 00 00	FF FF FF FF	à" ÿÿÿÿ
14:5FF0h:	FF FF FF FF	01 00 00 00	C8 1E 00 00	28 04 00 00	ÿÿÿÿ . . . È . . . (. .
14:6000h:	FF FF FF FF	00 00 00 00	00 00 00 00	0E 00 00 00	ÿÿÿÿ
14:6010h:	24 00 00 00	00 00 00 00	08 00 00 00	31 32 30 30	\$ 1 2 0 0
14:6020h:	30 30 30 32	E0 FF FF FF	76 6B 07 00	24 00 00 00	0002 à ÿÿ ÿv k . . \$. .
14:5FD0h:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00
14:5FE0h:	00 00 00 00	00 00 00 00	12 00 00 00	A 8 FF FF FF ÿÿÿÿ
14:5FF0h:	6E 4B 20 00	20 6E BB 46	6D 7E CD 01	00 00 00 00	nk . n>Fm~ Í
14:6000h:	40 1B 00 00	00 00 00 00	00 00 00 00	FF FF FF FF	@ ÿÿÿÿ
14:6010h:	FF FF FF FF	01 00 00 00	58 0D 00 00	50 0C 00 00	ÿÿÿÿ X . . . P . .
14:6020h:	FF FF FF FF	00 00 00 00	00 00 00 00	0E 00 00 00	ÿÿÿÿ

FIGURE 5: Image of pagefile.sys hex values while mounting a virtual disk using Encase (top) and FTK Imager (bottom).

Conflict of Interests

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

Acknowledgment

This research was supported by the Public Welfare & Safety Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2012M3A2A1051106).

References

- [1] M. Taylor, J. Haggerty, D. Gresty, and D. Lamb, "Forensic investigation of cloud computing systems," *Network Security*, vol. 2011, no. 3, pp. 4–10, 2011.
- [2] H. Chung, J. Park, S. Lee, and C. Kang, "Digital forensic investigation of cloud storage services," *Digital Investigation*, vol. 9, no. 2, pp. 81–95, 2012.
- [3] J. Dykstra and A. T. Sherman, "Design and implementation of FROST: digital forensic tools for the OpenStack cloud computing platform," *Digital Investigation*, vol. 10, pp. S87–S95, 2013.
- [4] A. Huth and J. Cebula, *The Basics of Cloud Computing*, Burlington, 2011.
- [5] P. Mell and T. Grance, "The NIST definition of cloud computing," NIST Special Publication 800–145, 2011.
- [6] Y. Pan and J. Zhang, "Parallel programming on cloud computing platforms," *Journal of Convergence*, vol. 3, pp. 23–28, 2012.
- [7] S. Biggs and S. Vidalis, "Cloud computing: the impact on digital forensic investigations," *Internet Technology and Secured Transactions*, pp. 1–6, 2009.
- [8] B. Martini and K.-K. R. Choo, "An integrated conceptual digital forensic framework for cloud computing," *Digital Investigation*, vol. 9, no. 2, pp. 71–80, 2012.
- [9] M. Taylor, J. Haggerty, D. Gresty, and R. Hegarty, "Digital evidence in cloud computing systems," *Computer Law & Security Review*, vol. 26, no. 3, pp. 304–308, 2010.
- [10] T. Teraoka, "Organization and exploration of heterogeneous personal data collected in daily life," *Human-Centric Computing and Information Sciences*, vol. 2, article 1, 2012.
- [11] S. Silas, K. Ezra, and E. B. Rajsingh, "A novel fault tolerant service selection framework for pervasive computing," *Human-Centric Computing and Information Sciences*, vol. 2, pp. 1–14, 2012.
- [12] T. J. Bittman, "Top five private cloud computing trends, 2012," http://blogs.gartner.com/thomas_bittman/2012/03/22/top-five-private-cloud-computing-trends-2012/.
- [13] S. Thorpe, "Virtual machine history model framework for a data cloud digital investigation," *Journal of Convergence*, vol. 3, 2012.
- [14] X. Xie, H. Jiang, H. Jin, W. Cao, P. Yuan, and L. T. Yang, "Metis: a profiling toolkit based on the virtualization of hardware performance counters," *Human-Centric Computing and Information Sciences*, vol. 2, pp. 1–15, 2012.
- [15] EMC White Paper, "Sizing EMC VNX Series for VDI workload," EMC, 2012.
- [16] Citrix, "XenServer Citrix eDocs," 2012, <http://support.citrix.com/proddocs/topic/xenserver/xs-wrapper.html>.
- [17] VMware, "VMware View architecture planning," 2012, <http://pubs.vmware.com/view-50/topic/com.vmware.ICbase/PDF/view-50-architecture-planning.pdf>.
- [18] J. Dykstra and D. Riehl, "Forensic collection of electronic evidence from infrastructure-as-a-service cloud computing," *Richmond Journal of Law and Technology*, vol. 19, no. 1, pp. 1–47, 2012.

Research Article

Fall-Detection Algorithm Using 3-Axis Acceleration: Combination with Simple Threshold and Hidden Markov Model

Dongha Lim,¹ Chulho Park,¹ Nam Ho Kim,^{1,2} Sang-Hoon Kim,¹ and Yun Seop Yu¹

¹ Department of Electrical, Electronic and Control Engineering and IITC, Hankyong National University, 327 Chungang-no, Anseong, Gyeonggi-do 456-749, Republic of Korea

² Laon People Co. Ltd., B-402 Bundang Technopark, 255 Yatapnam-ro, Bundang-gu, Seongnam, Gyeonggi-do 463-760, Republic of Korea

Correspondence should be addressed to Yun Seop Yu; ysyu@hknu.ac.kr

Received 10 February 2014; Accepted 19 August 2014; Published 17 September 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Dongha Lim 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.

Falls are a serious medical and social problem among the elderly. This has led to the development of automatic fall-detection systems. To detect falls, a fall-detection algorithm that combines a simple threshold method and hidden Markov model (HMM) using 3-axis acceleration is proposed. To apply the proposed fall-detection algorithm and detect falls, a wearable fall-detection device has been designed and produced. Several fall-feature parameters of 3-axis acceleration are introduced and applied to a simple threshold method. Possible falls are chosen through the simple threshold and are applied to two types of HMM to distinguish between a fall and an activity of daily living (ADL). The results using the simple threshold, HMM, and combination of the simple method and HMM were compared and analyzed. The combination of the simple threshold method and HMM reduced the complexity of the hardware and the proposed algorithm exhibited higher accuracy than that of the simple threshold method.

1. Introduction

In the past decade, the population in the world has been increasingly aging [1]. Korea, for example, is rapidly changing into an “aging society.” The elderly, especially those above the age of 65, are exposed to falls owing to the deterioration of their physical functions [2]. When an elder person falls and becomes unconscious or is unable to move his/her body, he/she may succumb to the injuries that caused the fall [3]. Thus, research and development of a system that can automatically detect falls in the elderly or other patients has been actively studied [4–7].

Because of the expansion of the Internet in the 90s, it is now commonly referred to as the Internet of Things (IoT). The pervasive and seamless interaction among objects, sensors, and computing devices is an important concern of the IOT [8]. Smart embedded objects such as a fall-detection sensor with wireless communication [9] will also become an important part of the IoT.

The identified fall-detection systems can be classified primarily into two categories: context-aware systems and wearable devices [10–21]. Context-aware systems use devices such as cameras, floor sensors, infrared sensors, microphones, pyroelectric infrared (PIR) sensors, and pressure sensors, deployed in the environment, to detect falls [10–13]. Their principal advantage is that a person is not required to wear any special equipment. Wearable device-based approaches rely on clothing with embedded sensors to detect the motion and location of the body of the subject [14–27]. The advantages of wearable devices are the cost efficiency, ease of installation, setup, and operation of the design.

There are two main approaches (algorithms) to detect falls: simple threshold and machine learning methods. In the simple threshold method, threshold values of specific parameters calculated from sensor data such as 3-axial acceleration are used to detect a fall [14–20]. Automatic fall detection using a threshold-based method of single parameters, calculated using acceleration measured by an accelerometer, has a high



FIGURE 1: Photograph of the sensor node (80 mm \times 50 mm) for fall detection. (a) Front side and (b) back side.

sensitivity (about 100%); however it has a relatively low specificity [14, 15]. Automatic fall detection using multiple parameter combinations has a relatively high sensitivity (85.7%) and specificity (90.1%) [16]. Automatic fall detection using angular velocities measured using a gyroscope has a high sensitivity (100%) and specificity (97.5%) [18]. Further, automatic fall detection using multiple parameters that are calculated using the acceleration and angular velocities measured by an accelerometer and a gyroscope, respectively, has a high sensitivity (91%) and specificity (92%) [19]. They are simple to implement and their computation effort is minimal. However, they have a problem with the tolerance of individual behavior and are less accurate for detecting falls that occur. In the machine learning method, various types of fall and activity of daily living (ADL) patterns are trained by a learning algorithm and then an event is classified as a fall or ADL by applying it to an evaluation algorithm [21–27]. The machine learning methods include support vector machine (SVM) [21, 22], Gaussian distribution of clustered knowledge [23], decision tree [24], and hidden Markov model (HMM) [25–27]. The machine learning method is more sophisticated and leads to better detection rates with accuracy of over 95%. Unfortunately, it is difficult to implement the machine learning approach due to the heavy computational and resource requirements [4]. The combination of the two approaches for fall detection has not yet been investigated.

In this paper, a fall-detection algorithm using 3-axis acceleration is proposed. The fall-feature parameters, calculated from the 3-axis acceleration, are applied to a simple threshold method [20]. Then, the falls that are determined from the simple threshold are applied to the HMM [25–27] to distinguish between falls and ADLs. The results from a simple threshold, HMM, and the combination of the simple method and HMM are compared and analyzed.

2. Materials and Methods

A novel fall-detection algorithm using an acceleration sensor node is presented. Because the chest of the subject is near the body's center of gravity [28], the sensor node is attached with an elastic belt on the chest of the subject. The sensor

node, as shown in Figure 1, measures sensor data and sends them to the gateway (portable computer (PC)) using a ZigBee network processor. The software environment used in the experiment was Visual Studio 2008 and fall-detection code was written in the C language on a Windows XP PC.

2.1. Subjects and Testing Activities. Intentional falls were performed by six healthy volunteers: four male and two female subjects whose ages ranged from 20 to 50, height from 160 to 185 cm, and weight from 50 to 85 kg. The falls were performed using a mattress (thickness: 20 cm). Each subject performed 7 types of activity (three types of fall and four types of ADL) as follows:

- (i) ADL-a: walking,
- (ii) ADL-b: running,
- (iii) ADL-c: standing jumping,
- (iv) ADL-d: lying down and standing up from a bed,
- (v) Fall-a: falling forward over something,
- (vi) Fall-b: falling laterally by losing balance,
- (vii) Fall-c: sliding and falling backward.

A total of 320 ADLs and 240 falls were tested. The total number of each activity for subjects A, B, C, and D (age: 20s) was 15 and for subjects E (age: 50s) and F (age: 40s) was 10. The ADLs used in this study were activities that could cause high impact or abrupt changes in a person's movement.

2.2. Hardware Description. The fall-detection system implemented in this paper consisted of a sensor node with a 3-axis accelerometer ± 8 g triaxial accelerometer (BMA150, Bosch) [30] and wireless communication module (CC2530, Texas Instrument) [31], a gateway to collect the information from multiple wireless sensor nodes, and a server to determine falls by applying the parameters from the 3-axis acceleration to the proposed fall-detection algorithm. The sensor was controlled by the ZigBee network processor. The sampling rate was set to 100 Hz, a bandwidth exceeding the characteristic response of human movement. Each triaxial acceleration was statistically

calibrated in order to correct any possible axis tilt due to the orientation of the device on the subject or lower back tilt of the subject. To execute the algorithm that detects a fall suffered by an elder person, the gateway monitors the accelerations sent from the sensor nodes and calculates several parameters including the directions, magnitudes, and angles of the elder person's motion from the sensing data. The algorithm stores the measured data and calculates the parameters.

2.3. Fall-Feature Parameters. To detect a fall, five types of parameters are used in the analyses [27]. The fall-feature parameters of sum vector magnitude (SVM) A_{SVM} , differential SVM (DSVM) A_{DSVM} of acceleration, angle θ , gravity-weighted SVM (GSVM) A_{GSVM} , and gravity-weighted DSVM (GDSVM) A_{GDSVM} are calculated using the following equations [27]:

$$\begin{aligned}
 A_{SVM}(i) &= \sqrt{A_x^2(i) + A_y^2(i) + A_z^2(i)}, \\
 \theta(i) &= \tan^{-1} \left(\frac{\sqrt{A_y^2(i) + A_z^2(i)}}{A_x(i)} \right) \times \frac{180}{\pi}, \\
 A_{DSVM}(i) &= \left((A_x(i) - A_x(i-1))^2 + (A_y(i) - A_y(i-1))^2 \right. \\
 &\quad \left. + (A_z(i) - A_z(i-1))^2 \right)^{1/2}, \\
 A_{GSVM}(i) &= \frac{\theta(i)}{90} \times A_{SVM}(i), \\
 A_{GDSVM}(i) &= \frac{\theta(i)}{90} \times A_{DSVM}(i),
 \end{aligned} \tag{1}$$

where i denotes the sample number and $A_x(i)$, $A_y(i)$, and $A_z(i)$ denote the x -axial, y -axial, and z -axial accelerations of the i th sample, respectively. The Euler angle θ denotes the tilted angle between the accelerometer y -axis and the vertical direction.

2.4. Fall-Detection Algorithm. Figure 2 shows the flow diagram for the fall-detection system. Real-time 3-axis accelerations are sent from the sensor handset of the subject to the server through the ZigBee network and then the five types of fall-feature parameters are calculated from the sample data in the learning and evaluating range.

The fall-feature parameters are applied to the simple threshold method to determine whether a parameter is above a certain threshold within a time interval. If any parameter is above a threshold, the sample is determined to be a possible fall indicating a subject fall event or ADL similar to a fall event. The simple threshold algorithm for multiple parameters using double parameters is shown in Algorithm 1. The thresholds are determined from the receiver operating characteristics (ROC) curve from which both true positive

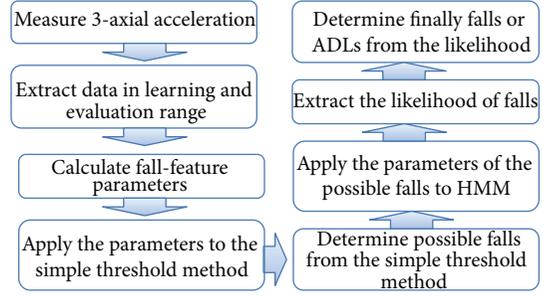


FIGURE 2: Flow diagram for falls detection system.

and false positive rates are calculated for all parameter values. The threshold values are determined when the specificity is best with a sensitivity of 100%. Instead of using all the events, only the fall-feature parameters of the events determined to be possible falls from the simple threshold method are applied to the HMM algorithm [29, 32–34] as shown in Figure 3 and Algorithm 2. First, the learning process of the HMM is performed for four types of ADL and three types of fall; all the values in the model matrices $\lambda_i = (A_i, B_i, \pi_i)$ of all the single parameters for i activities of four types of ADL and three types of fall are calculated using Baum-Welch learning algorithm [29]. A_i , B_i , π_i , M , and N denote the state transition probability distribution, observation emission probability distribution, initial state distribution, number of invisible states, and number of observation values for i activities, respectively. In this paper, M and N are used as 4 and 8, respectively. Then, based on the leaning database, an activity is evaluated by applying the HMM with the parameter. The likelihood of all the single parameters for the observation sequences is calculated using the HMM evaluation algorithm [29]. Finally the maximum probability among the four types of ADL and three types of fall is determined. If the selected activity with the maximum probability is among the three types of fall, the fall is alarmed; otherwise, it is determined to be an ADL.

3. Experimental Results

Figure 4 shows the ROC curve of the fall detection obtained from the simple threshold using a single parameter. The single parameter $A_{GSVM} = 2.5$ g has the best fall detection with sensitivity, specificity, and accuracy of 92.92, 81.56, and 88.05%, respectively. The best specificity with 100% sensitivity is 68.13% when $\theta = 60^\circ$. This is chosen as the threshold value.

Figures 5, 6, 7, and 8 show the ROC curves of the fall detection obtained from the simple threshold using the double parameters of A_{SVM} and θ , A_{DSVM} and θ , A_{GSVM} and θ , and A_{GDSVM} and θ , respectively, as shown in Algorithm 1. Each best fall detection using double parameters is shown in the captions of Figures 5, 6, 7, and 8. As the false positive rate is decreased, the true positive rate is abruptly decreased. Among them, the best fall detection is of sensitivity 98.75%, specificity 94.38%, and accuracy 96.25% when $A_{SVM} = 2.5$ g and $\theta = 65^\circ$, as shown in Figure 5. The best specificity with 100% sensitivity is 91.56% and accuracy is 95.18% when

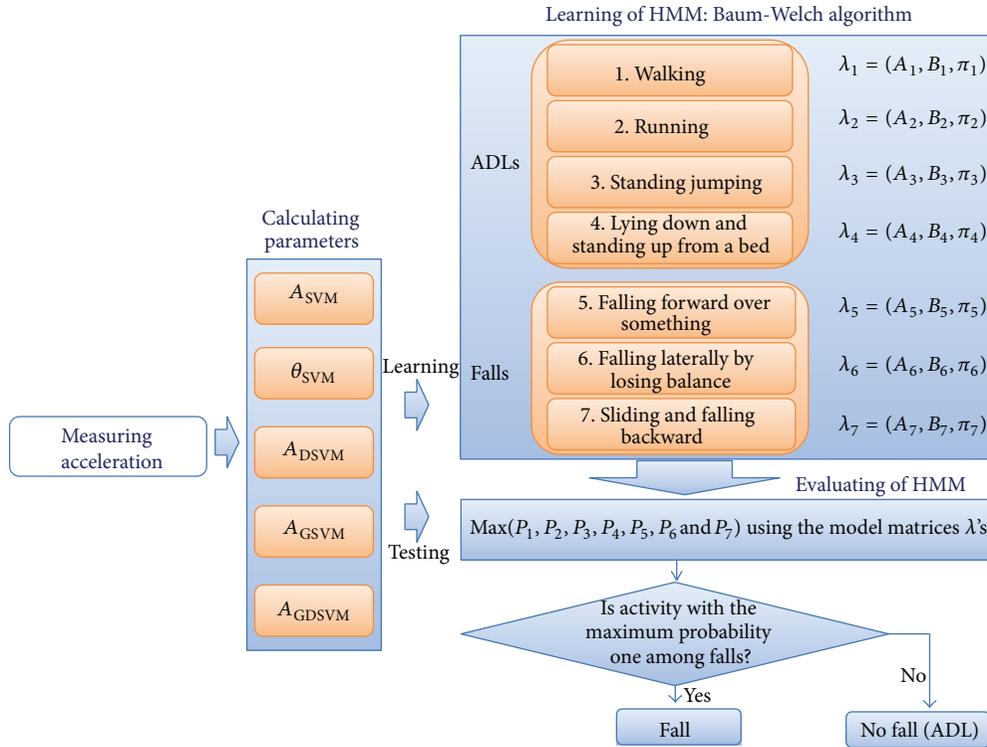


FIGURE 3: Configuration of fall-detection system applying HMM.

- ```

(1) if the parameter > threshold value of the parameter
 then
(2) if θ > threshold value of θ
 (among 100 samples after satisfying the condition in Line 1)
 then
(3) return fall detection
(4) return no fall detection

```

ALGORITHM 1: Simple threshold algorithm using double parameters for fall detection.

$A_{GSVM} = 2.5 \text{ g}$  and  $\theta = 55^\circ$ . These are chosen as threshold values. Table 1 shows falls and ADLs determined with these threshold values. In this table, the total numbers of each activity for subjects A, B, C, and D and subjects E and F are 15 and 10, respectively. All events of the three types of fall are detected as a fall, but 27 events of ADLs are detected as falls instead of ADLs. Subjects C, D, E, and F especially failed to detect several lying-down events (ADL-4) as ADLs, and subjects E and F of over 40 years old also failed to detect a few running and standing jump events as ADLs. It shows that the simple threshold method has a limitation to detect lying-down events of all subjects and running and jumping events of relatively old subjects.

Table 2 shows the falls and ADLs in which the fall events (267 events) chosen from the simple threshold method using double parameters as shown in Table 1 are evaluated by applying the parameter  $\theta$  to the HMM as shown in Algorithm 2, which is the best of fall and ADL detection results applying 5 types of parameters to HMM as shown in Table 3. Instead

of evaluating all 560 events, only the fall events (267 events) chosen from the simple threshold method using double parameters are applied only to the HMM. Computing effort and resources can be saved, compared to applying all the events to the HMM. The sensitivity, specificity, and accuracy obtained from applying the parameter  $\theta$  to the HMM are 99.17%, 99.69%, and 99.5%, respectively. One lying-down event of subject E (over 40 years old) is detected as a fall instead of an ADL, and one forward fall and one backward fall events of subject F (over 50 years old) are detected as ADLs instead of falls. The experimental results of combining the simple threshold with the HMM are higher than those with the simple threshold method only.

#### 4. Conclusions

To detect falls, the fall-detection algorithm combining a simple threshold method and an HMM with 3-axis acceleration was proposed. To apply the proposed fall-detection algorithm

- (1) Calculate all the values in model matrices  $\lambda_i = (A_i, B_i, \pi_i)^*$  of all single parameters for  $i$  activities of four types of ADL and three types of fall by the Baum-Welch learning algorithm [29]
  - (2) Calculate all the likelihoods of all single parameters for observation sequences using the evaluation algorithm [29]
  - (3) Find a maximum probability among four types of ADL and three types of fall
  - (4) **If** the activity with the maximum probability is among three types of fall **then** fall detection
  - (5) **Else** no fall (ADL) detection
- \*  $A_i, B_i,$  and  $\pi_i$  denote the state transition probability distribution, observation emission probability distribution, and initial state distribution for  $i$  activities, respectively.

ALGORITHM 2: HMM algorithm for fall detection.

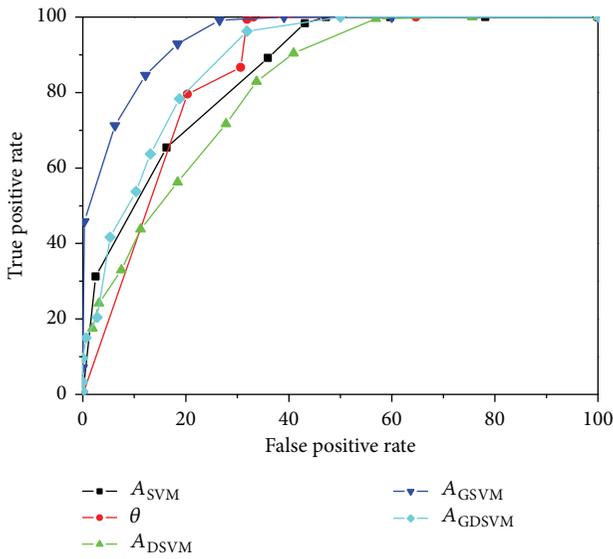


FIGURE 4: ROC curve of fall detection obtained from the simple threshold using the single fall-feature parameter. The best fall detection is of sensitivity 92.92%, specificity 81.56%, and accuracy 88.05% when  $A_{GSVM} = 2.5$  g.

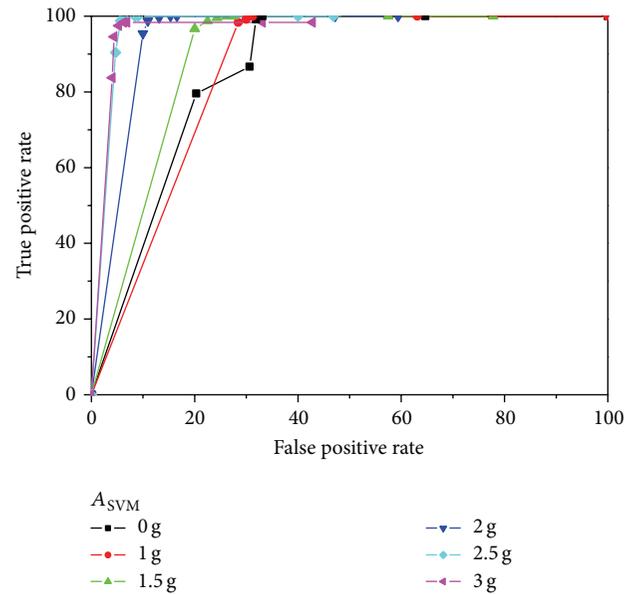


FIGURE 5: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{SVM}$  and  $\theta$ . The best fall detection is of sensitivity 98.75%, specificity 94.38%, and accuracy 96.25% when  $A_{SVM} = 2.5$  g and  $\theta = 65^\circ$ .

and detect falls, a wearable fall-detection device was designed and produced. The several fall-feature parameters of the 3-axis acceleration were introduced and applied to the simple threshold method. Possible falls were chosen through the simple threshold and then applied to the HMM to solve the problems such as deviation of interpersonal falling behavioral patterns and similar fall actions. The double parameters  $A_{SVM} = 2.5$  g and  $\theta = 65^\circ$  showed the best fall detection with sensitivity, specificity, and accuracy of 98.75%, 94.38%, and 96.25%, respectively. The best fall detection combining the simple threshold and HMM was of sensitivity 99.17%, specificity 99.69%, and accuracy 99.5% when the threshold values for the simple threshold method were  $A_{SVM} = 2.5$  g and  $\theta = 55^\circ$  and the parameter  $\theta$  was applied to the HMM. These results are higher than those with the simple threshold method using double parameters. Applying only the fall events determined from the simple threshold method to the HMM reduced the computing effort and resources, compared to those of using all the events applied to the HMM. Because the proposed algorithms are simple, they can be implemented into an

TABLE 1: Best fall and ADL detection results obtained from the simple threshold method with the double threshold values  $A_{SVM} = 2.5$  g and  $\theta = 55^\circ$  (sensitivity = 100% and specificity = 91.56% and accuracy = 95.18%).

| Subjects (ages) | ADL-a | ADL-b | ADL-c | ADL-d | Fall-a | Fall-b | Fall-c |
|-----------------|-------|-------|-------|-------|--------|--------|--------|
| A (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| B (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| C (20s)         | 15    | 15    | 15    | 11    | 15     | 15     | 15     |
| D (20s)         | 15    | 15    | 15    | 11    | 15     | 15     | 15     |
| E (40s)         | 10    | 10    | 5     | 7     | 10     | 10     | 10     |
| F (50s)         | 10    | 4     | 6     | 9     | 10     | 10     | 10     |

embedded system such as an 8051-based microcontroller with 128 Kbyte ROM. In the future, if the proposed algorithms are implemented to the embedded system, its performance will be tested in a real time.

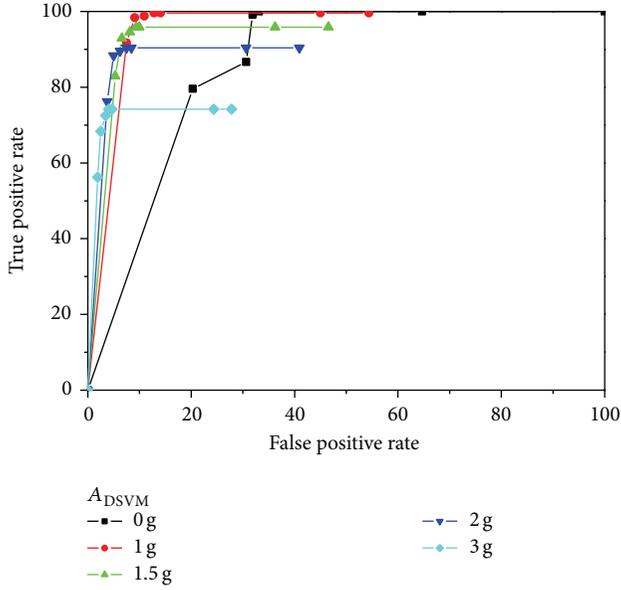


FIGURE 6: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{DSVM}$  and  $\theta$ . The best fall detection is of sensitivity 98.33%, specificity 90.94%, and accuracy 95.16% when  $A_{DSVM} = 1g$  and  $\theta = 65^\circ$ .

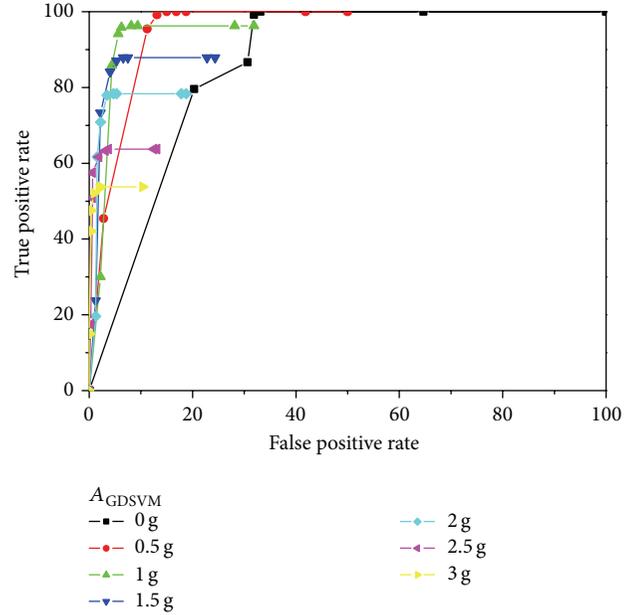


FIGURE 8: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{GDSVM}$  and  $\theta$ . The best fall detection is of sensitivity 95.83%, specificity 93.75%, and accuracy 94.94% when  $A_{GDSVM} = 1g$  and  $\theta = 60^\circ$ .

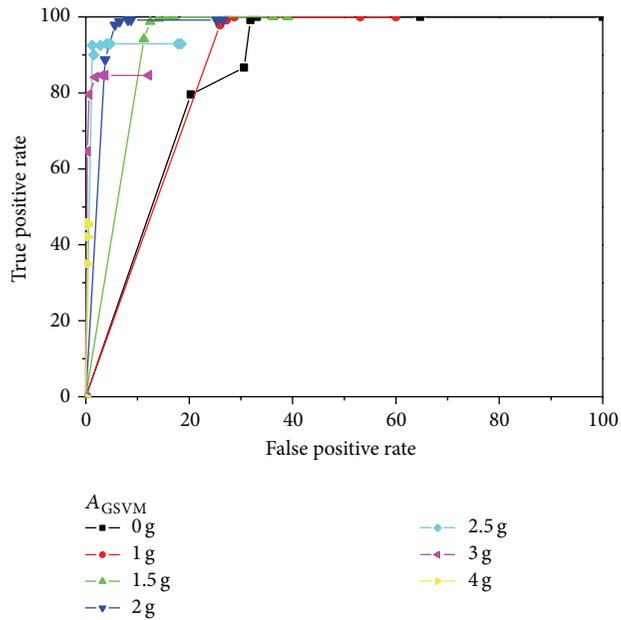


FIGURE 7: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{GSVM}$  and  $\theta$ . The best fall detection is of sensitivity 97.92%, specificity 94.38%, and accuracy 96.4% when  $A_{GSVM} = 2g$  and  $\theta = 65^\circ$ .

One limitation of this study is that the fall-detection algorithm was tested on subjects in the age group 20–50, falling under simulated conditions. Further research is required to test the performance of the proposed algorithm for detecting actual falls suffered by the elderly.

TABLE 2: Best fall and ADL detection results evaluated by applying the parameter  $\theta$  of the observation samples determined with fall events shown in Table 1 to the HMM algorithm shown in Algorithm 2 (sensitivity = 99.17% and specificity = 99.69% and accuracy = 95.89%).

| Subjects (ages) | ADL-a | ADL-b | ADL-c | ADL-d | Fall-a | Fall-b | Fall-c |
|-----------------|-------|-------|-------|-------|--------|--------|--------|
| A (20s)         | 15    | 15    | 15    | 9     | 15     | 15     | 15     |
| B (20s)         | 15    | 15    | 15    | 4     | 15     | 15     | 15     |
| C (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| D (20s)         | 15    | 15    | 15    | 12    | 15     | 15     | 15     |
| E (40s)         | 10    | 10    | 10    | 9     | 10     | 10     | 10     |
| F (50s)         | 10    | 10    | 10    | 10    | 9      | 10     | 9      |

TABLE 3: Fall detecting results evaluated by combination of the simple threshold method with double parameters and HMM.

|             | $\theta$     | $A_{SVM}$ | $A_{DSVM}$ | $A_{GSVM}$ | $A_{GDSVM}$ |
|-------------|--------------|-----------|------------|------------|-------------|
| Sensitivity | <b>99.17</b> | 97.5      | 99.6       | 99.17      | 99.17       |
| Specificity | <b>99.69</b> | 95.63     | 97.81      | 96.88      | 97.5        |
| Accuracy    | <b>99.5</b>  | 96.43     | 98.57      | 97.86      | 98.21       |

### 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 GRRC program of Gyeonggi Province [(GRRC HANKYONG 2012-B02), Development of

Vision Inspection algorithm and Wireless and Wired Integrated Control System for Intelligent Logistics Inspection].

## References

- [1] C. A. Werner, "The Older Population: 2010," Census Briefs U.S. Bureau of the Census, 2010, <http://www.census.gov/prod/cen2010/briefs/c2010br-09.pdf>.
- [2] B. M. H. Park, J. C. Ha, I. H. Shin et al., "Senior survey 2008: life and welfare service needs of the elderly in Korea," Ministry for Health and Welfare, 2009.
- [3] B. Kaluža and M. Luštrek, "Fall detection and activity recognition methods for the confidence project: a survey," in *Proceedings of the 12th International Multiconference Information Society*, vol. A, pp. 22–25, 2008.
- [4] R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *BioMedical Engineering Online*, vol. 12, no. 1, article 66, 2013.
- [5] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.
- [6] R. Hegde, B. G. Sudarshan, S. C. P. Kumar, S. A. Hariprasad, and B. S. Satyanarayana, "Technical advances in fall detection system—a review," *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 7, pp. 152–160, 2013.
- [7] S.-R. Ke, H. L. U. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, "A review on video-based human activity recognition," *Computers*, vol. 2, pp. 88–131, 2013.
- [8] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [9] J. K.-Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.
- [10] B. Töreyn, Y. Dedeoglu, and A. Cetin, "HMM based falling person detection using both audio and video," in *Computer Vision in Human-Computer Interaction*, pp. 211–220, Springer, Berlin, Germany, 2005.
- [11] Y. Li, K. C. Ho, and M. Popescu, "A microphone array system for automatic fall detection," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291–1301, 2012.
- [12] X. Luo, T. Liu, J. Liu, X. Guo, and G. Wang, "Design and implementation of a distributed fall detection system based on wireless sensor networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2012, article 118, 2012.
- [13] H. Rimminen, J. Lindström, M. Linnavuo, and R. Sepponen, "Detection of falls among the elderly by a floor sensor using the electric near field," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 6, pp. 1475–1476, 2010.
- [14] M. Kangas, I. Vikman, J. Wiklander, P. Lindgren, L. Nyberg, and T. Jämsä, "Sensitivity and specificity of fall detection in people aged 40 years and over," *Gait & Posture*, vol. 29, no. 4, pp. 571–574, 2009.
- [15] P.-K. Chao, H.-L. Chan, F.-T. Tang, Y.-C. Chen, and M.-K. Wong, "A comparison of automatic fall detection by the cross-product and magnitude of tri-axial acceleration," *Physiological Measurement*, vol. 30, no. 10, pp. 1027–1037, 2009.
- [16] A. Weiss, I. Shimkin, N. Giladi, and J. M. Hausdorff, "Automated detection of near falls: algorithm development and preliminary results," *BMC Research Notes*, vol. 3, article 62, 2010.
- [17] A. K. Bourke, J. V. O'Brien, and G. M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm," *Gait and Posture*, vol. 26, no. 2, pp. 194–199, 2007.
- [18] A. K. Bourke and G. M. Lyons, "A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor," *Medical Engineering & Physics*, vol. 30, no. 1, pp. 84–90, 2008.
- [19] Q. Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach, and G. Zhou, "Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information," in *Proceedings of the 6th International Workshop on Wearable and Implantable Body Sensor Networks (BSN '09)*, pp. 138–143, Berkeley, Calif, USA, June 2009.
- [20] Y. J. Yi and Y. S. Yu, "Emergency-monitoring system based on newly-developed fall detection algorithm," *Journal of Information and Communication Convergence Engineering*, vol. 11, no. 3, pp. 147–154, 2013.
- [21] T. Zhang, J. Wang, L. Xu, and P. Liu, "Fall detection by wearable sensor and one-class SVM algorithm," in *Intelligent Computing in Signal Processing and Pattern Recognition*, vol. 345 of *Lecture Notes in Control and Information Science*, pp. 858–863, 2006.
- [22] C. Doukas, I. Maglogiannis, P. Tragas, D. Liapis, and G. Yovanof, "Patient fall detection using support Vector Machines," *International Federation for Information Processing*, vol. 247, pp. 147–156, 2007.
- [23] M. Yuwono, B. D. Moulton, S. W. Su, B. G. Celler, and H. T. Nguyen, "Unsupervised machine-learning method for improving the performance of ambulatory fall-detection systems," *BioMedical Engineering Online*, vol. 11, article 9, 11 pages, 2012.
- [24] H. Kerdegari, K. Samsudin, A. R. Ramli, and S. Mokaram, "Evaluation of fall detection classification approaches," in *Proceedings of the 4th International Conference on Intelligent and Advanced Systems (ICIAS '12)*, pp. 131–136, Kuala Lumpur, Malaysia, June 2012.
- [25] J. Cheng, X. Chen, and M. Shen, "A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals," *IEEE Journal on Biomedical and Health Informatics*, vol. 17, no. 1, pp. 38–45, 2013.
- [26] L. Tong, Q. Song, Y. Ge, and M. Liu, "HMM-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.
- [27] N. H. Kim and Y. S. Yu, "Fall recognition algorithm using gravity-weighted 3-axis accelerometer data," *Journal of the Institute of Electronics and Information Engineers*, vol. 50, no. 6, pp. 254–259, 2013.
- [28] M. Kangas, A. Konttila, P. Lindgren, I. Winblad, and T. Jämsä, "Comparison of low-complexity fall detection algorithms for body attached accelerometers," *Gait & Posture*, vol. 28, no. 2, pp. 285–291, 2008.
- [29] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [30] "BMA150 Triaxial acceleration sensor Data sheet," Bosch Sensortec, [http://ae-bst.resource.bosch.com/media/products/dokumentation/bma150/bma150\\_flyer\\_rev13\\_14jan2008\\_redlich.pdf](http://ae-bst.resource.bosch.com/media/products/dokumentation/bma150/bma150_flyer_rev13_14jan2008_redlich.pdf).
- [31] CC2530: A True System-on-Chip Solution for 2.4-GHz IEEE 802.15.4 and ZigBee Applications, Texas Instruments Incorporated.
- [32] M. A. Fattah, "The use of MSVM and HMM for sentence alignment," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 301–314, 2012.

- [33] W. O. Odoyo, J.-H. Choi, I.-K. Moon, and B.-J. Cho, "Silhouette-edge-based descriptor for human action representation and recognition," *Journal of Information and Communication Convergence Engineering*, vol. 11, no. 2, pp. 124–131, 2013.
- [34] A. Hussain, A. R. Abbasi, and N. Afzulpurkar, "Detecting & interpreting self-manipulating hand movements for student's affect prediction," *Human-Centric Computing and Information Sciences*, vol. 2, article 14, 2012.

## Research Article

# Linear SVM-Based Android Malware Detection for Reliable IoT Services

**Hyo-Sik Ham,<sup>1</sup> Hwan-Hee Kim,<sup>1</sup> Myung-Sup Kim,<sup>2</sup> and Mi-Jung Choi<sup>1</sup>**

<sup>1</sup> Department of Computer Science, Kangwon National University, 1 Kangwondaehak-gil, Gangwon-do 200-701, Republic of Korea

<sup>2</sup> Department of Computer and Information Science, Korea University, 2511 Sejong-ro, Sejong-si 339-770, Republic of Korea

Correspondence should be addressed to Mi-Jung Choi; mjchoi@kangwon.ac.kr

Received 31 January 2014; Accepted 22 July 2014; Published 3 September 2014

Academic Editor: Young-Sik Jeong

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

Current many Internet of Things (IoT) services are monitored and controlled through smartphone applications. By combining IoT with smartphones, many convenient IoT services have been provided to users. However, there are adverse underlying effects in such services including invasion of privacy and information leakage. In most cases, mobile devices have become cluttered with important personal user information as various services and contents are provided through them. Accordingly, attackers are expanding the scope of their attacks beyond the existing PC and Internet environment into mobile devices. In this paper, we apply a linear support vector machine (SVM) to detect Android malware and compare the malware detection performance of SVM with that of other machine learning classifiers. Through experimental validation, we show that the SVM outperforms other machine learning classifiers.

## 1. Introduction

The Internet of Things (IoT) is the communications between things or physical and logical objects organized with networks to extend into a communication network like the existing Internet [1]. It is a generic term of technologies that have intelligent interfaces which actively interact. If things communicate with each other and have intelligent interfaces, they would have new functions beyond their own existing characteristics. The newly obtained properties would bring us convenience and huge usefulness. Machine-to-machine communication or IoT is likely to serve a company with the advancement of smartphones.

IoT technologies and smartphones have been connected to provide a variety of services all over the world. Audi, a German company, offers a service that automatically records data such as mileage and location of electric bicycles through a smartphone [2], while TBWA Helsinki, a company in the Republic of South Africa, provides a service that connects smartphones with a shop window outside a store to check and purchase goods by touching the show window [3]. NEC in Japan installs sensors measuring conditions such

as temperature, humidity, and rainfall on a farm to enable smartphones to manage the farmland and crops [4]. Lockitron, an American company, provides a door lock service using smartphones without keys [5]. Likewise, most IoT services are monitored and controlled through smartphone applications.

By combining IoT with smartphones, many convenient IoT services [6] have been provided to users. For example, using smartphone's range of sensors (accelerometer, Gyro, video, proximity, compass, GPS, etc.) and connectivity options (cell, WiFi, Bluetooth, NFC, etc.), we can have a well-equipped Internet of Things device in our pocket that can automatically monitor our movements, location, and workouts throughout the day. The Alohar Mobile Ambient Analytics Platform [7] efficiently collects location and other mobile sensor data and quickly analyzes it to understand a smartphone user's behavior. Through smartphone applications, we can remotely monitor and manage your home and cut down on your monthly bills and resource usage. Smart thermostats like the Nest [8] use sensors, real-time weather forecasts, and the actual activity in your home during the day to reduce your monthly energy usage by up to 30%. We can

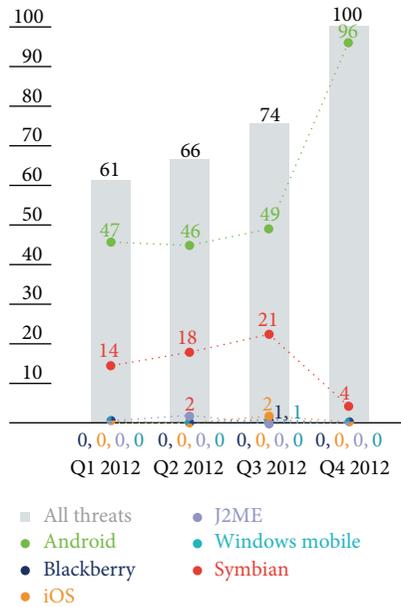


FIGURE 1: Increase of Android malware.

use the Nest application to connect to the thermostat from a smartphone and can change the temperature miles from home. However, there are adverse underlying effects in this scenario such as invasion of privacy and information leakage. Because there is a diversity of important personal information such as user's location, contact information, and certificates in a smartphone, hackers pose a serious threat [9, 10]. Currently, hackers are expanding their targets from existing PCs to smartphones. Security measures should be prepared to protect users against attacks. A method should be prepared as a security mechanism for detecting and controlling malware that leaks information from smartphones or causes malicious damage through malfunction [11].

Figure 1 is a report by Finnish security company F-secure which states that, 301 mobile malware samples from 2012, 238 samples targeted the Android platform [23]. While the amount of malware that targeted other mobile platforms gradually decreased as time went on from the 1st to 4th quarter, Android showed a contrasting result. The reason for the increase in Android malware was its open source policy and its leniency to market application verification. In addition, it easily allowed the distribution of malware in the market through the repackaging method of inserting it in a normal application.

Previous studies showed various approaches to detecting mobile malware such as signature-based detection [12–15], behavior-based detection [16–20], and taint analysis-based detection [21, 22]. This paper identifies the issues of previous studies and proposes a detection method through a linear support vector machine (SVM) [24] to secure reliable IoT services [25]. The linear SVM shows high performance among machine learning algorithms in order to effectively detect malware in the Android platform with monitored resources during application runtime.

The organization of this paper is as follows. In Section 2, we summarize previous studies on mobile malware detection and briefly introduce the linear SVM algorithm as a related work. In Section 3, we explain resource monitoring information and system for detecting malware. In Section 4, we show experimental results for malware detection using various machine learning classifiers. In Section 5, we conclude this paper and propose possible future work.

## 2. Related Works

This section examines the trends of previous studies and explains the linear SVM method for detecting mobile malware.

*2.1. Mobile Malware Detection Trends.* To detect abnormal behaviors occurring in an existing mobile environment (malware, virus, worm, etc.), signature-based detection, behavior-based detection, and taint analysis-based detection were performed. Trends of the studies are summarized in Table 1 based on their detection techniques and collected data.

Signature-based detection [12–15] is a traditional method used to detect malware in a PC environment. To define signature, static and dynamic methods are simultaneously used. Static analysis targets the source and object codes and analyzes the codes without actually starting a program. It decompiles the source code of a malware to discover vulnerabilities that occur in commands, statements, and so on. Dynamic analysis is a method of finding certain patterns in memory leakage, traffic flow, and data flow while actually running the program. However, a large amount of storage is required for applying this method to the mobile environment, and the performance overhead is high for pattern matching.

Behavior-based detection [16–20] is a method of detecting invasion status by comparatively analyzing predetermined attack patterns and process behavior that occur in a system. It is one of the studies that has been receiving the most attention recently due to signature-based detection's limited detection of malicious behavior. To detect abnormal patterns, it mainly monitors event information that occurs in smartphone features such as memory usage, SMS content, and battery consumption. Host-based detection (for directly monitoring information inside a device) and network-based detection (for gathering information via network) are frequently used. Since host-based detection increases the usage of a smartphone's battery and memory, a detection method of collecting data inside the device and transmitting the data to an outside analysis server is mainly used. In addition, a machine learning technique is used to improve the analysis rate of dynamic data. Therefore, it is highly important to choose the proper features to be collected and select a suitable machine learning algorithm for accurate detection.

Dynamic analysis-based detection [21, 22], also called "taint analysis," is a method of marking specific data and monitoring the process of data being sent in an application code to track the flow of data. Since a smartphone runs in a virtual machine, this method is considered appropriate. However, it is no longer being studied due to the difficulty

TABLE 1: Trends of studies on mobile malware detection techniques.

| Detection technique        | Author               | Collected data           | Description                                                                                                                                                                                                      |
|----------------------------|----------------------|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Signature-based technique  | Schmidt et al. [12]  | Executable file analysis | Uses the readelf command to carry out static analysis on executable files using system calls                                                                                                                     |
|                            | Bläsing et al. [13]  | Source code analysis     | Uses the Android sandbox to carry out static/dynamic analysis on applications                                                                                                                                    |
|                            | Kou and Wen [14]     | Packet analysis          | Uses functions such as packet-preprocessing and pattern-matching to detect malware                                                                                                                               |
|                            | Bose et al. [15]     | API call history         | Collects system events of upper layers and monitors their API calls to detect malware                                                                                                                            |
| Behavior-based technique   | Schmidt et al. [16]  | System log data          | Detects anomalies in terms of Linux kernels and monitors traffic, kernel system calls, and file system log data by users                                                                                         |
|                            | Cheng et al. [17]    | SMS, Bluetooth           | Lightweight agents operating in smartphones record service activities such as usage of SMS or Bluetooth, comparing the recorded results with users' average values to analyze whether there is intrusion or not. |
|                            | Liu et al. [18]      | Battery consumption      | Monitors abnormal battery consumption of smartphones to detect intrusion by newly created or currently known attacks                                                                                             |
|                            | Burguera et al. [19] | System call              | Monitors system calls of smartphone kernel to detect external attacks through outsourcing                                                                                                                        |
|                            | Shabtai et al. [20]  | Process information      | Continuously monitors logs and events and classifies them into normal and abnormal information                                                                                                                   |
| Dynamic analysis technique | Fuchs et al. [21]    | Data marking             | Analyzes malware by carrying out static taint analysis for Java source code                                                                                                                                      |
|                            | William et al. [22]  | Data marking             | Modifies stack frames to add taint tags into local variables and method arguments and traces the propagation process through tags to analyze malware                                                             |

in applying it in an actual environment and because of the overhead of tracking data flow to a low level.

**2.2. Malware Detection via Linear SVM.** In this paper, malware is detected based on the collected data by monitoring resources in an Android environment. Behavior-based detection involves the inconvenience of having to determine malware infection status by examining numerous features. Accordingly, behavior-based detection uses a machine learning method to enable automated malware classification and to ensure its identification and accuracy. The machine learning method is a method of entering the data collected from the device as learning data to create a learning model and applying some of the other data to the learning model.

A diversity of classifiers is used for machine learning techniques. Typically, there are DT (decision tree), BN (Bayesian networks), NB (naive Bayesian), Random forest, and SVM (support vector machine). DT [26] is a tree for sorting based on the feature value to classify instances. In this way, it calculates probability values of being able to reach each node and draws a result depending on the probability values. BN [27] is a graphic model that combines a probability theory based on Bayesian theory with a graphic theory. In other words, it makes a conditional probability table with the given data and configures a topology of the graph to draw a conclusion. NB [28] assumes dependent features as independent ones and calculates their probabilities to draw a conclusion. RF [29] combines decision trees formed by the independently sampled random vectors to draw a conclusion

and shows a relatively higher detection rate. RF is a machine learning classifier frequently used for malware detection studies in the Android environment [30, 31]. Neural networks technique [32] is another machine learning technique. However, because neural networks technique consumes more time than other classifiers when training [33], it is considered difficult to apply to the malware detection system in which real time is emphasized. Therefore, this paper does not consider neural networks.

In this paper, a linear SVM method [24] is applied to detect malware. SVM is one of the machine learning classifiers receiving the most attention currently, and its various applications are being introduced because of its high performance [34]. The SVM could also solve the problem of classifying nonlinear data. Of the input features, unnecessary ones are removed by the SVM machine learning classifier itself and the modeling is carried out, so there is some overhead in the aspect of time. However, it could be expected to perform better than other machine learning classifiers in the aspect of complexity or accuracy in analysis [35].

Figure 2 shows how to find hyperplanes which are criteria for the SVM to do the learning process to classify data. All hyperplanes (a), (b) and (c) classify two things correctly, but the greatest advantage of the SVM is that it selects hyperplane (c) which maximizes the margin (the distance between data) and accordingly maximizes the capability of generalization. Therefore, even if input data is located near a hyperplane, it has an advantage of being able to classify more correctly compared to other classifiers. We verify that

TABLE 2: Selected features for malware detection.

| Resource type | Resource feature                                            |                                                                                                            |
|---------------|-------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| Network       | RxBytes, TxBytes, RxPacket, TxPacket                        |                                                                                                            |
| Telephone     | Send/receive call                                           |                                                                                                            |
| SMS Message   | Send/receive SMS                                            |                                                                                                            |
| CPU           | CPU usage                                                   |                                                                                                            |
| Battery       | Level, temperature, voltage                                 |                                                                                                            |
| Process       | Process ID, process name, running process, context switches |                                                                                                            |
| Memory        | Native                                                      | Total size, shared size, allocated size, physical page, virtual set size, free size, heap size, dirty page |
|               | Dalvik                                                      | Total size, shared size, allocated size, physical page, virtual set size, free size, heap size, dirty page |

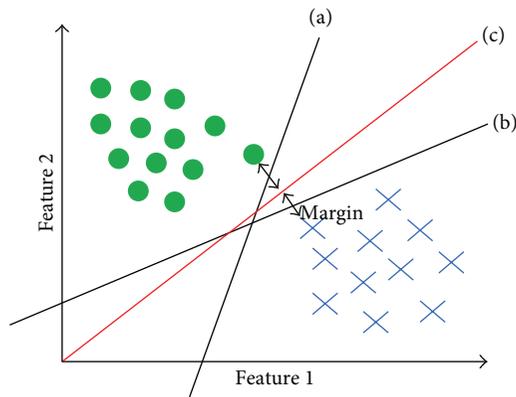


FIGURE 2: Data classification method of SVM.

SVM shows a good detection performance by comparing the experimental results of SVM with those of other machine learning classifiers (Bayesian network, decision tree, naïve Bayesian, and random forest) and SVM analysis technique.

### 3. Collection of Resource Information for Malware Detection

This section presents a method of collecting resource information for detecting Android malware. It explains collected resource features and agents designed and implemented to collect resource information inside Android devices.

**3.1. Resource Features for Malware Detection.** For detecting malware that is the target of analysis, resource information generated in a device is monitored when a user executes normal applications or abnormal applications infected with malware. In a previous study [20], every resource and event generated in an Android device was defined and all these features were used for analyzing malware.

However, the number of features is 88, which are too many, most of them having low correlation, with the Android memory structure not being reflected. In addition, some of these 88 features could be extracted only if the root permission is acquired. The 32 features proposed in this paper are information that could be extracted even without the root permission. In this paper, 32 features that are highly related to targeted malware, as shown in Table 2, are defined by

classifying them into seven categories according to resource type. This study does not monitor the total memory usage that simply changes through an application execution but monitors the usage amount classified into native area and Dalvik machine area by considering the memory characteristics of the Android platform. Dalvik machine memory is allocated when running each application.

For the features proposed in this paper, every feature was extracted about network, phone, message, CPU, battery, and memory for each process. The existing study [20] used a feature selection algorithm such as the information gain to increase the detection system's performance, but this paper did not carry out the feature selection. As also mentioned in Section 2.2, the reason was because the SVM classifier autonomously carried out dimensional reduction function to use only the required features for determining results.

**3.2. Malware Detection System Architecture.** To monitor the selected resource features, an agent is needed that can continuously monitor the corresponding features inside a device. This experiment alternatively executes a normal application and an abnormal application on the Android platform to test malware detection. Figure 3 shows the structure of the Android malware detection system, which primarily consists of a mobile agent and an analysis server.

First, the mobile agent collects information for each application through the resource monitoring component. The data is collected from the Linux kernel in the mobile agent, and the feature extractor is responsible for the collecting of actual data. The feature extractor is comprised of four collectors, and they collect information on variations in network, memory, CPU, and battery. The collected feature information is specified in Table 2 of Section 3.1. This collected information is transferred to the data management module, and the data management module transforms the collected information into a vector form. The data constructed as a vector form by the data management module is transferred to the analysis server for evaluation. At such time, the reason for transmitting data to an external server is because its overhead is large in the aspect of time and resource if the modeling and analysis are carried out by machine learning in the mobile device. Therefore, to minimize such an overhead, malware detection is carried out in the analysis server, and only the detection result is transferred again to the mobile agent.

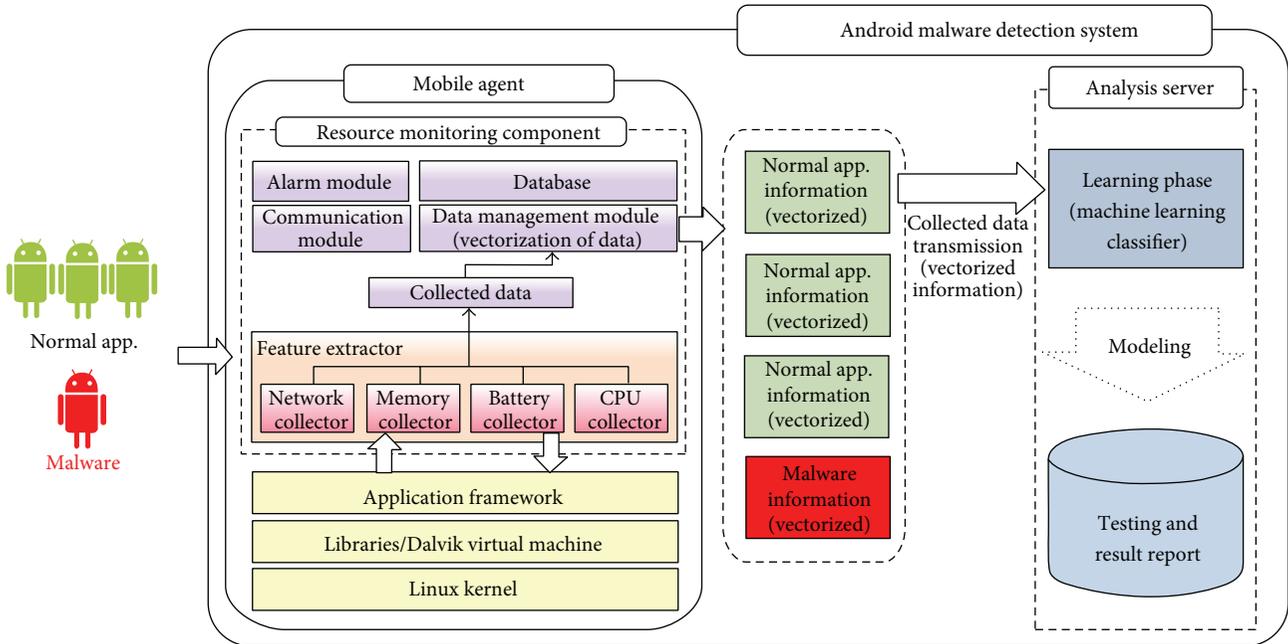


FIGURE 3: Android malware detection system architecture.

The analysis server learns by using the vectorized resource data for each application, which is transferred from the mobile agent as input data. After learning, a model (pattern) of the resource data for each application is created and, based on it, the existence of malware is determined. If malware is detected, an alarm message is transmitted to the user through the alarm module. Figure 4 represents the algorithm of the malware detection system proposed in this paper as a sequence diagram.

Examining the overall flow of the algorithm, it extracts feature information for each application and makes the machine learning classifier to learn the extracted information. Based on this learned information, it determines the existence of malware. This method is not much different from existing malware detection studies. Upon comparing this paper with the existing studies, however, a difference is found on the information on features and the applied machine learning classifier.

#### 4. Experimental Results

This section applies the proposed linear SVM technique. It demonstrates the superiority of the SVM by comparing it with four machine learning classifiers and describes the experimental methods and results.

**4.1. Android Malware Categories.** This study chooses 14 of the latest malware programs for each category to verify the proposed method. Malicious applications are selected on the basis of the “typical cases of malware causing great damage to users” presented in the 2012 ASEC report [36] from Ahnlab in Korea. Most of the Android-targeted malware is divided into Trojan, spyware, root permission acquisition (exploit),

and installer (dropper). The reason for Trojan having a large proportion of the selected malware is because most of the malicious codes that occurred in 2012 were Trojan. Table 3 describes the malware to be analyzed in this study.

- (i) Trojan: it looks harmless, but it is a program containing a risk factor in effect. Malware is usually included in the program, so it basically executes the malware when running the application.
- (ii) Spyware: a compound word formed from “spy” and “software” and it is a type of malware that is secretly installed on a device to collect information. It is frequently used for commercial uses such as repeatedly opening pop-up advertisement or redirecting users to a particular website. Spyware causes inconvenience by changing a device’s settings or being difficult to delete.
- (iii) Root permission acquisition (exploit): it uses unknown vulnerabilities or 0-day attacks. The new vulnerability is discovered but not yet patched for. It is malware that acquires root permission to clear security settings and makes additional attacks on the Android platform.
- (iv) Installer (dropper): it conceals malware in a program and guides users to run malware and spyware. These days, because it does not install one kind of malware but multiple ones with the advent of multidroppers, it makes detection more difficult.

**4.2. Elements of Data Set.** This paper uses 14 normal applications and 14 malicious ones embedded with malware to test malware detection. The data set is composed of 90% normal and 10% malicious applications. The reason for composing

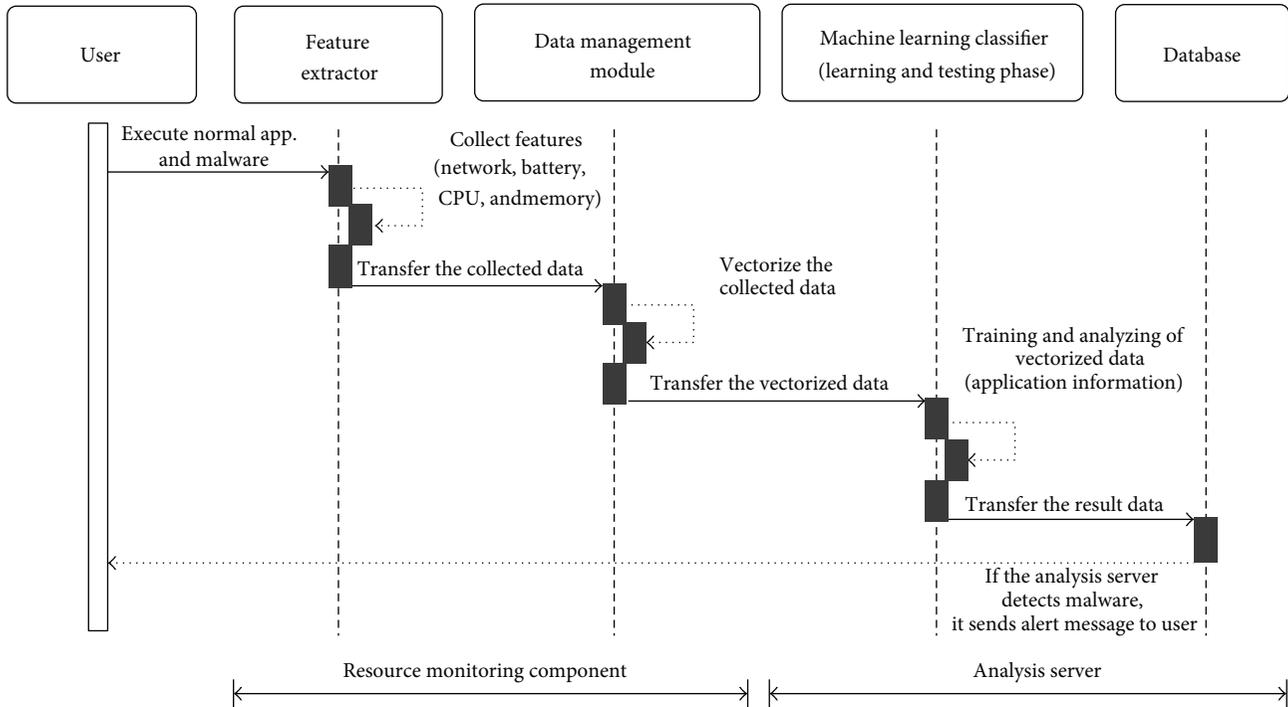


FIGURE 4: Sequence diagram for malware detection system.

TABLE 3: Features of malware to be analyzed.

| Malware category                      | Malware Name | Features                                                                |
|---------------------------------------|--------------|-------------------------------------------------------------------------|
| Trojan                                | Zitmo        | Disguises as an Android security application                            |
|                                       | DroidKungFu  | Leaks personal information                                              |
|                                       | Opfake       | Disguises as a game application (performance degradation)               |
|                                       | FakeInst     | Disguises as a game application (performance degradation)               |
|                                       | Goldream     | Disguises as a game/animation application                               |
|                                       | LightDD      | Disguise as an adult application                                        |
| Spyware                               | Geimini      | Carries out a backdoor function                                         |
|                                       | Adrd.AQ      | Carries out a backdoor function                                         |
|                                       | Snake        | Disguises as a game to leak information                                 |
|                                       | Pjapps       | Adds malicious functions to a normal app.                               |
| Root permission acquisition (exploit) | Rooror.BT    | Makes terminal rooting (security dismantling)                           |
|                                       | Basebridge   | Acquires root permissions and then communicates with an external server |
| Installer (dropper)                   | SMSHider     | Guides to install malware through SMS                                   |
|                                       | Anserver     | Downloads other malware                                                 |

the data set in this way is that normal applications are more common than malicious ones when examining the ratio of applications used in the actual mobile environment. In experiment, we construct the data set using a 5-fold cross-validation method.

Figure 5 shows the 5-fold cross-validation method applied to the data collected from respective devices. As shown in Figure 5, the data collected from other devices are crossed to organize the training and test sets. If the dataset is organized like this, all the collected data are organized as the training and test sets, so it could be said that it is a method

considering portability between devices. In other words, it shows that malware detection is possible even if the device's environment is different. It could also be verified that the selected features are useful for detecting malware.

**4.3. Evaluation Indicators.** This section describes evaluation indicators to verify the performance of experimental results. The indicators used in this paper are TPR (true positive rate), FPR (false positive rate), precision, accuracy, and  $F$ -measure. Statistical information for the decision result is

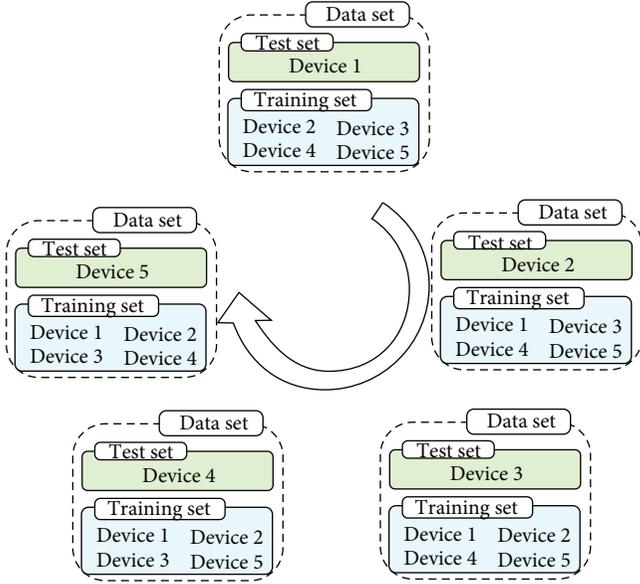


FIGURE 5: Composition of training and test data set.

required to find the respective evaluation indicators. Table 4 is a confusion matrix for computing the evaluation indicators.

TP (true positive) is a numerical value of identifying the uninfected status of a normal application. TN (true negative) represents a number that correctly identifies an application containing malware. FN (false negative) means a number that incorrectly finds malware in an actually normal application. FP (false positive) represents a number that incorrectly finds no malware despite an application actually containing malware. Based on the statistical information above, this paper finds TPR (true positive rate), FPR (false positive rate), precision, accuracy, and *F*-measure. Equations (1)–(5) for respective indicators are as follows:

$$TPR = \frac{TP}{TP + TN}, \quad (1)$$

$$FPR = \frac{FP}{FP + TN}, \quad (2)$$

$$Precision = \frac{TP}{FP + FP}, \quad (3)$$

$$Accuracy = \frac{TP}{TP + FP}, \quad (4)$$

$$F\text{-measure} = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}. \quad (5)$$

True positive rate (TPR) represents the proportion (1) of correctly identified normal applications. False positive rate (FPR) represents the proportion (2) of malware-containing applications incorrectly identified as safe. If applications containing malware are misdiagnosed, they could cause serious damage to the system, so this indicator is considered important. Precision is an indicator representing an error of the decision value, which represents the proportion (3) of correctly diagnosed normal applications. Accuracy is an

TABLE 4: Confusion matrix of evaluation indicators.

|             |          | Predicted data      |                     |
|-------------|----------|---------------------|---------------------|
|             |          | Positive            | Negative            |
| Actual data | Positive | TP (true positive)  | FN (false negative) |
|             | Negative | FP (false positive) | TN (true negative)  |

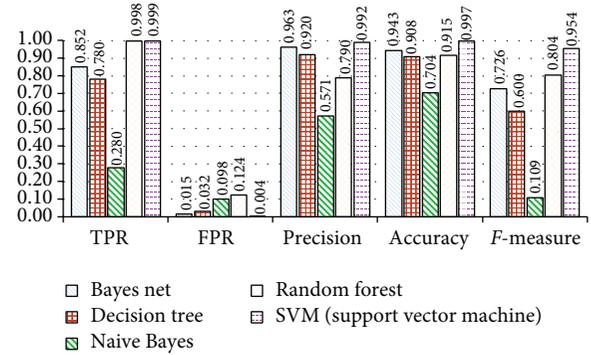


FIGURE 6: Detection results of respective classifiers.

indicator representing the system’s accuracy, expressed in the proportion (4) of correctly identified normal applications and ones containing malware, respectively, among the results. *F*-measure is also called *F1*-score and means accuracy (5) in the aspect of decision results.

**4.4. Experimental Results.** Figure 6 shows malware detection results according to machine learning classifiers. From the TPR perspective, the random forest (TPR = 0.998) and SVM (TPR = 0.999) show a good performance. For the FPR used as the most important evaluation indicator when detecting malware, SVM has FPR = 0.004, which could be determined as the best classifier because its ratio of incorrectly classifying normal applications as malicious is small, and it shows far better performance than other classifiers also in terms of accuracy and precision.

Table 5 shows the results of the detailed malware detection of respective classifiers’ TPR/FPR indicators. RF has Adrd.AQ (TPR = 1.000), Anserver (TPR = 0.996), and Geimini (TPR = 0.962), which show higher performance than other classifiers. For other malware, however, it is shown that SVM gives higher performance with TPR = 0.953 on average. In particular, NB does not at all detect specific malware (Adrd.AQ, Anserver, DroidKungFu, GoldDream, Opfake, PjApps, SMSHider, and Snake). For Opfake, SVM gives relatively lower performance with TPR = 0.820. The reason is that Opfake is expanded from FakeInst, which shows similar patterns, so it incorrectly detects Opfake as FakeInst. However, it shows that TPR is about 31% more improved than the random forest. Every classifier shows a low numerical value in terms of FPR, but upon analysis of the correlation with TPR it could be found that SVM shows the best performance. Because the NB classifier’s TPR is also 0.000 if its FPR is 0.000, it could be said that NB is a classifier unsuitable for detecting malware.

TABLE 5: Detailed performance indicators of machine learning classifiers (TPR/FPR).

| Normal and malware | TPR   |       |       |              |              | FPR   |              |              |              |              |
|--------------------|-------|-------|-------|--------------|--------------|-------|--------------|--------------|--------------|--------------|
|                    | BN    | DT    | NB    | RF           | SVM          | BN    | DT           | NB           | RF           | SVM          |
| Normal             | 0.852 | 0.780 | 0.280 | 0.998        | <b>0.999</b> | 0.015 | 0.032        | 0.098        | 0.124        | <b>0.004</b> |
| Adrd.AQ            | 0.695 | 0.671 | 0.000 | <b>1.000</b> | 0.957        | 0.012 | 0.017        | <b>0.000</b> | 0.004        | 0.002        |
| Anserver           | 0.985 | 0.918 | 0.000 | <b>0.996</b> | 0.957        | 0.051 | 0.117        | <b>0.000</b> | 0.004        | <b>0.000</b> |
| Basebridge         | 0.692 | 0.862 | 0.487 | 0.671        | <b>0.939</b> | 0.009 | 0.056        | 0.081        | 0.014        | <b>0.000</b> |
| DroidKungFu        | 0.720 | 0.868 | 0.000 | 0.874        | <b>0.977</b> | 0.008 | <b>0.000</b> | <b>0.000</b> | <b>0.000</b> | 0.001        |
| FakeInst           | 0.946 | 0.709 | 0.263 | 0.838        | <b>0.985</b> | 0.005 | <b>0.000</b> | 0.001        | 0.001        | 0.011        |
| Geimini            | 0.649 | 0.464 | 0.000 | <b>0.962</b> | 0.893        | 0.004 | 0.009        | <b>0.000</b> | <b>0.000</b> | 0.001        |
| GoldDream          | 0.567 | 0.298 | 0.000 | 0.717        | <b>0.994</b> | 0.012 | 0.005        | <b>0.000</b> | 0.022        | 0.002        |
| LightDD            | 0.663 | 0.562 | 0.373 | 0.645        | <b>0.957</b> | 0.012 | 0.035        | 0.284        | <b>0.000</b> | <b>0.000</b> |
| Opfake             | 0.567 | 0.429 | 0.000 | 0.509        | <b>0.820</b> | 0.005 | 0.002        | <b>0.000</b> | 0.001        | 0.005        |
| PjApps             | 0.946 | 0.659 | 0.000 | 0.548        | <b>0.996</b> | 0.032 | 0.012        | <b>0.000</b> | 0.003        | 0.003        |
| RouterBT           | 0.868 | 0.451 | 0.782 | 0.573        | <b>0.966</b> | 0.009 | <b>0.000</b> | 0.318        | 0.008        | 0.004        |
| SMSHider           | 0.778 | 0.766 | 0.000 | 0.773        | <b>0.949</b> | 0.001 | 0.054        | <b>0.000</b> | 0.001        | 0.001        |
| Snake              | 0.422 | 0.205 | 0.000 | 0.703        | <b>0.935</b> | 0.013 | 0.007        | <b>0.000</b> | 0.001        | 0.001        |
| Zitmo              | 0.750 | 0.503 | 0.378 | 0.789        | <b>0.967</b> | 0.060 | 0.025        | 0.087        | 0.033        | <b>0.001</b> |
| Average            | 0.740 | 0.610 | 0.171 | 0.773        | <b>0.953</b> | 0.017 | 0.025        | 0.058        | 0.014        | <b>0.002</b> |

TABLE 6: Detailed performance indicators of machine learning classifiers (precision/accuracy/ $F$ -measure).

| Normal and malware | Precision    |              |       |              |              | Accuracy     |       |       |              |              | $F$ -measure |       |       |       |              |
|--------------------|--------------|--------------|-------|--------------|--------------|--------------|-------|-------|--------------|--------------|--------------|-------|-------|-------|--------------|
|                    | BN           | DT           | NB    | RF           | SVM          | BN           | DT    | NB    | RF           | SVM          | BN           | DT    | NB    | RF    | SVM          |
| Normal             | 0.963        | 0.920        | 0.571 | 0.790        | <b>0.992</b> | 0.943        | 0.908 | 0.704 | 0.915        | <b>0.997</b> | 0.904        | 0.844 | 0.375 | 0.882 | <b>0.995</b> |
| Adrd.AQ            | 0.682        | 0.590        | 0.000 | 0.893        | <b>0.939</b> | 0.978        | 0.972 | 0.964 | <b>0.996</b> | <b>0.996</b> | 0.689        | 0.628 | 0.000 | 0.943 | <b>0.948</b> |
| Anserver           | 0.532        | 0.315        | 0.000 | 0.933        | <b>0.993</b> | 0.951        | 0.885 | 0.945 | 0.996        | <b>0.997</b> | 0.691        | 0.469 | 0.000 | 0.963 | <b>0.975</b> |
| Basebridge         | 0.803        | 0.455        | 0.246 | 0.724        | <b>0.999</b> | 0.976        | 0.940 | 0.897 | 0.970        | <b>0.997</b> | 0.744        | 0.596 | 0.327 | 0.696 | <b>0.968</b> |
| DroidKungFu        | 0.842        | <b>1.000</b> | 0.000 | 0.997        | 0.983        | 0.977        | 0.993 | 0.945 | 0.993        | <b>0.998</b> | 0.776        | 0.929 | 0.000 | 0.932 | <b>0.980</b> |
| FakeInst           | 0.910        | <b>1.000</b> | 0.911 | 0.973        | 0.836        | <b>0.992</b> | 0.985 | 0.960 | 0.990        | 0.989        | <b>0.928</b> | 0.830 | 0.408 | 0.900 | 0.905        |
| Geimini            | 0.842        | 0.607        | 0.000 | <b>0.996</b> | 0.957        | 0.986        | 0.976 | 0.971 | <b>0.999</b> | 0.995        | 0.733        | 0.526 | 0.000 | 0.979 | <b>0.924</b> |
| GoldDream          | 0.730        | 0.780        | 0.000 | 0.653        | <b>0.962</b> | 0.964        | 0.956 | 0.945 | 0.963        | <b>0.997</b> | 0.639        | 0.431 | 0.000 | 0.683 | <b>0.978</b> |
| LightDD            | 0.765        | 0.481        | 0.070 | 0.997        | <b>0.998</b> | 0.971        | 0.943 | 0.697 | 0.981        | <b>0.998</b> | 0.710        | 0.518 | 0.118 | 0.783 | <b>0.977</b> |
| Opfake             | 0.878        | 0.910        | 0.000 | <b>0.979</b> | 0.900        | 0.972        | 0.966 | 0.945 | 0.972        | <b>0.985</b> | 0.689        | 0.583 | 0.000 | 0.670 | <b>0.858</b> |
| PjApps             | 0.554        | 0.707        | 0.000 | 0.880        | <b>0.941</b> | 0.967        | 0.975 | 0.959 | 0.978        | <b>0.997</b> | 0.699        | 0.682 | 0.000 | 0.675 | <b>0.967</b> |
| RouterBT           | 0.846        | <b>1.000</b> | 0.117 | 0.802        | 0.926        | 0.985        | 0.972 | 0.687 | 0.971        | <b>0.994</b> | 0.857        | 0.621 | 0.203 | 0.669 | <b>0.946</b> |
| SMSHider           | <b>0.983</b> | 0.451        | 0.000 | 0.972        | 0.976        | 0.987        | 0.936 | 0.946 | 0.986        | <b>0.996</b> | 0.868        | 0.568 | 0.000 | 0.861 | <b>0.962</b> |
| Snake              | 0.651        | 0.646        | 0.000 | <b>0.987</b> | 0.977        | 0.955        | 0.949 | 0.945 | 0.983        | <b>0.995</b> | 0.512        | 0.312 | 0.000 | 0.821 | <b>0.956</b> |
| Zitmo              | 0.325        | 0.439        | 0.144 | 0.479        | <b>0.977</b> | 0.933        | 0.958 | 0.893 | 0.960        | <b>0.998</b> | 0.454        | 0.469 | 0.209 | 0.596 | <b>0.972</b> |
| Average            | 0.754        | 0.687        | 0.137 | 0.870        | <b>0.957</b> | 0.969        | 0.954 | 0.894 | 0.977        | <b>0.995</b> | 0.726        | 0.600 | 0.109 | 0.804 | <b>0.954</b> |

Table 6 shows the detailed results of respective classifiers' precision/accuracy. For the decision tree, the precision of DroidKungFu, FakeInst, and RouterBT is 1.000, which marks the best performance. Their average precision is 0.687, which is lower than SVM (precision = 0.957). For accuracy, SVM shows higher performance with 0.995 on average. For the  $F$ -measure, it is found that the SVM is 0.954 on average except for FakeInst, which gives superior performance from other classifiers.

## 5. Conclusion and Future Work

This paper proposed an Android malware-detection mechanism using machine learning algorithms for reliable IoT services. This paper also proposed a machine learning technique to remedy the disadvantage of the behavior-based technique (one of the mobile detection techniques) and to correctly detect malware targeting the Android platform. The first problem of existing studies was that they were not suitable

for generalization because they were not able to analyze many types of malware. To solve this problem, the recent domestic trend of malware targeting Androids was evaluated and 14 malware programs were selected to apply them to the proposed method. Second, because the features of the existing papers focused only on the detection of some types of malware or they had no correlation with malware, their detection rate was reduced. This paper reflected the structural characteristics of the Android platform to subdivide its memory space. This study also selected the features having much correlation with malware to increase efficiency. Third, the portability between devices was considered to verify it through the 5-fold cross-validation experimental method. We concluded that the SVM technique could accurately detect most malware in a relative sense by comparatively analyzing them with four classifiers (Bayesian network, decision tree, naïve Bayesian, and random forest).

Future studies may consider exposing hardly detectable malware by resource information and sharper system accuracy. Because diverse variants and new types of mobile malware are on the rise, further study on a technique that could detect future malware should be scheduled. We plan to develop an efficient and lightweight implementation of the SVM algorithm that can be embedded to a smartphone for real-time detection. We also plan to conduct malware elimination and control by applying detection results to actual mobile devices and networks.

## Conflict of Interests

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

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2013RIA1A3011698).

## References

- [1] N. Gershenfeld, R. Krikorian, and D. Cohen, "The internet of things," *Scientific American*, vol. 291, no. 4, pp. 76–81, 2004.
- [2] Readwrite, "The Future of Connected Cars: What Audi is Driving Towards," 2012, <http://readwrite.com>.
- [3] Brandingmagazine, *Adidas' Interactive Window Shopping Experience*, 2012, <http://www.brandingmagazine.com/>.
- [4] K. Moessner, F. Le Gall, P. Cousin et al., "Internet of things strategic research and innovation agenda," in *Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems*, 2013.
- [5] J. Jensen, S. Copenhagen, and A. H. Larsen, *Smart Intercom-Enhancing the Apartment Intercom System*, ACM Computing Classification System, 2012.
- [6] T.-T. Truong, M.-T. Tran, and A.-D. Duong, "Improvement of the more efficient & secure ID-based remote mutual authentication with key agreement scheme for mobile devices on ECC," *Journal of Convergence*, vol. 3, no. 2, pp. 25–36, 2012.
- [7] Aloha, "Mobile Ambient Analytics Platform," <https://www.alohar.com/developer/>.
- [8] Nest, "Nest Thermostat," <https://nest.com/>.
- [9] D. Werth, A. Emrich, and A. Chapko, "An ecosystem for user-generated mobile service," *Journal of Convergence*, vol. 3, no. 4, 2012.
- [10] J. W. K. Gnanaraj, K. Ezra, and E. B. Rajsingh, "Smart card based time efficient authentication scheme for global grid computing," *Human-centric Computing and Information Sciences*, vol. 3, article 16, 2013.
- [11] K. Sakurai and K. Fukushima, "Actual condition and issues for mobile security system," *Journal of Information Processing Systems*, vol. 3, no. 2, pp. 54–63, 2007.
- [12] A. Schmidt, A. Camtepe, and S. Albayrak, "Static smartphone malware detection," in *Proceedings of the 5th Security Research Conference (Future Security 2010)*, p. 146, 2010.
- [13] T. Bläsing, L. Batyuk, A.-D. Schmidt, S. A. Camtepe, and S. Albayrak, "An android application sandbox system for suspicious software detection," in *Proceedings of the 5th International Conference on Malicious and Unwanted Software (Malware '10)*, pp. 55–62, Nancy, France, October 2010.
- [14] X. Kou and Q. Wen, "Intrusion detection model based on android," in *Proceedings of the 4th IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT '11)*, pp. 624–628, October 2011.
- [15] A. Bose, X. Hu, K. G. Shin, and T. Park, "Behavioral detection of malware on mobile handsets," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 225–238, June 2008.
- [16] A.-D. Schmidt, H.-G. Schmidt, J. Clausen et al., "Enhancing security of linux-based android devices," in *Proceedings of the 15th International Linux Kongress*, Lehmann, October 2008.
- [17] J. Cheng, S. H. Y. Wong, H. Yang, and S. Lu, "SmartSiren: virus detection and alert for smartphones," in *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (MobiSys '07)*, pp. 258–271, June 2007.
- [18] L. Liu, G. Yan, X. Zhang, and S. Chen, "VirusMeter preventing your cellphone from spies," in *Recent Advances in Intrusion Detection*, vol. 5758 of *Lecture Notes in Computer Science*, pp. 244–264, 2009.
- [19] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani, "Crowdroid behavior-based malware detection system," in *Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM '11)*, 2011.
- [20] A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer, and Y. Weiss, "'Andromaly': a behavioral malware detection framework for android devices," *Journal of Intelligent Information Systems*, vol. 38, no. 1, pp. 161–190, 2012.
- [21] A. P. Fuchs, A. Chaudhuri, and J. S. Foster, "Scan-Droid Automated Security Certification of Android Applications," 2011.
- [22] E. William, P. Gilbert, C. Byung-Gon et al., "TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones," in *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation (OSDI '10)*, pp. 1–6, USENIX Association, Berkeley, Calif, USA.
- [23] F-Secure, "Mobile Threat Report," Q4, 2012.
- [24] C. J. C. Burgesm, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

- [25] C. M. Medaglia and A. Serbanati, "An overview of privacy and security issues in the internet of things," in *The Internet of Things*, pp. 389–395, Springer, New York, NY, USA, 2010.
- [26] S. B. Kotsiantis, "Supervised machine learning: a review of classification techniques," *Informatica*, vol. 31, no. 3, pp. 249–268, 2007.
- [27] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997.
- [28] R. Kohavi, *Scaling up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid*, KDD, 1996.
- [29] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [30] H. S. Ham and M. J. Choi, "Analysis of Android malware detection performance using machine learning classifiers," in *Proceedings of the International Conference on ICT Convergence (ICTC '13)*, pp. 490–495, 2013.
- [31] T. Kim, Y. Choi, S. Han et al., "Monitoring and detecting abnormal behavior in mobile cloud infrastructure," in *Proceedings of the IEEE Network Operations and Management Symposium (NOMS '12)*, pp. 1303–1310, April 2012.
- [32] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*, vol. 1, Pws, Boston, Mass, USA, 1996.
- [33] S. Mukkamala, G. Janoski, and A. Sung, "Intrusion detection using neural networks and support vector machines," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '02)*, vol. 2, pp. 1702–1707, IEEE, May 2002.
- [34] Y. Hwang, J. Kwon, J. Moon, and S. Cho, "Classifying malicious web pages by using an adaptive support vector machine," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 39–404, 2013.
- [35] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [36] Ahnlab, "Ahnlab ASEC Report," 2012.

## Research Article

# SCondi: A Smart Context Distribution Framework Based on a Messaging Service for the Internet of Things

**Jongmoon Park and Myung-Joon Lee**

*Department of Electrical/Electronic and Computer Engineering, University of Ulsan, 93 Daehak-ro, Nam-gu, Ulsan 680-749, Republic of Korea*

Correspondence should be addressed to Myung-Joon Lee; mjlee@ulsan.ac.kr

Received 25 March 2014; Accepted 6 May 2014; Published 26 August 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 J. Park and M.-J. Lee. 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.

When developing IoT (Internet of Things) applications, context distribution is a key feature to support effective delivery of related contextual data obtained from things to all interested entities. With the advent of the IoT era, multiple billion devices can generate huge amounts of data that might be used in IoT applications. In this paper, we present a context distribution framework named SCondi utilizing the messaging service which supports MQTT—an OASIS standard IoT messaging protocol. SCondi provides the notion of *context channel* as a core feature to support efficient and reliable mechanism for distributing huge context information in the IoT environment. The context channel provides a pluggable filter mechanism that supports effective extraction, tailoring, authentication, and security of information.

## 1. Introduction

Context-aware computing, as a component of a ubiquitous computing [1–4], is a core technology that supports human-centric intelligent service using contextual information in real situations. As machine-centric world (M2M, machine-to-machine) [5, 6] is coming, the notion of context awareness supporting improved response ability plays an important role in the IoT since things connected to the internet have awareness/sensing ability in many cases [7–9].

The IoT which provides useful services combined with various IT technologies is considered to have the potential to change our world. The IoT is a future internet environment defined as a dynamic global network infrastructure [10]. Things with identities and intelligent interfaces can be active participants wherever they are enabled to interact and communicate. To support useful services based on the massive information related to these things, several technologies such as REST (representational state transfer) [11], MQTT (message queuing telemetry transport) [12, 13], XMPP (extensible messaging and presence protocol) [14], and CoAP (constrained application protocol) [15] are being studied for information delivery, message process, and communication protocol. Recently, the MQTT protocol has been

adopted by OASIS (organization for the advancement of structured information standards) as a standard messaging protocol for the IoT.

With the advent of the IoT era, multiple billion devices can generate huge amounts of data that might be used in IoT applications. However, it is not an easy challenge to deliver these tremendous data at the right time, to the right place, and with the right quality. As one of the important components in context-aware computing, context distribution can be a key feature to support effective delivery of contextual data in the IoT environment [16]. As of now, although many research works on context distribution has been conducted, there is no standard mechanism for reliable and sophisticated context distribution in the IoT environment.

In this paper, we propose a context distribution framework named SCondi utilizing the messaging service which supports MQTT. SCondi provides the notion of *context channel* as a core feature to support efficient and reliable mechanism for distributing huge context information in the IoT environment. The context channel is an abstract communication channel which can reliably tailor and disseminate a collection of information to service providers. Based on the MQTT messaging service, the context channel provides a pluggable filter mechanism that supports effective extraction,

tailoring, authentication, and security of information. In short, the context channel provides higher level abstract mechanism for information delivery as a transport medium of context-aware systems in the IoT environment.

## 2. Backgrounds

IoT is a future internet environment that focuses on machine to machine communication, referring to uniquely identifiable objects and their virtual representations. MQTT is a standard Internet of Things connectivity protocol that is designed as an extremely lightweight publish/subscribe messaging transport, considering limited computing power and poor network connectivity. The MQTT protocol is developed by IBM and chosen as standard by OASIS.

As HTTP has made a web to be an infrastructure that share information over the internet, MQTT is expected to be a key infrastructure that makes billions of embedded low price devices to go online. It is already widely used in a lot of embedded systems.

Mosquitto is an open source (BSD licensed) message broker that implements the MQTT 3.1, providing a lightweight method of carrying out messaging using a publish/subscribe model [17, 18]. Mosquitto is designed to fit messaging among machines such as low-power sensors, mobile devices, embedded computers, and microcontrollers, supporting various OS platforms such as Microsoft's Windows, Apple's OS X, and Linux family.

Context-aware computing is a core technology that supports human-centric intelligent service using contextual information in real situations. Context refers to a variety of information that can define the state of the real world's entities, generally consisting of information such as entity identity, activity, status, time, and location [19]. Since many devices which can sense contextual information are getting connected to the internet, the context-aware computing plays an important role in the IoT.

## 3. Design of SCondi

In this section, we introduce a context distribution framework named SCondi which is based on a messaging service for the IoT, supporting smart and effective dissemination of context data. We begin by mentioning the core requirements for the context data distribution proposed by Bellavista et al. [20]. To satisfy the requirements, the context channel is provided with filter mechanism as a key facility for reliable context data distribution.

*3.1. Key Requirements of Context Data Distribution.* Bellavista et al. mentioned that context data distribution needs to meet following 5 requirements for providing an effective context-aware service in IoT environment.

- (1) Communication should be asynchronous and anonymous among context producers and consumers.
- (2) To support mobile heterogeneous and wireless scenarios, the context data distribution has to promptly adapt to mobility and current available resources.

- (3) The context data distribution must enforce visibility scopes of context data to avoid useless management overhead.
- (4) The context data distribution has to enforce QoC-based constraints for timeliness and reliability guarantees of data delivery.
- (5) The context data distribution has to handle data life cycle for self-control of the distribution process.

To satisfy the requirements, SCondi provides the following features. First, adopting MQTT as a messaging mechanism, our framework supports asynchronous and anonymous communication among message publishers and subscribers (satisfying (1)). Secondly, our framework provides effective mobility and reliable message delivery based on MQTT's QoS (Quality of Service) in limited network environments (satisfying (2)). Thirdly, the context channel in our framework provides filter chain mechanism for the QoC constraints such as context data management, resource access control, data validation, and timeliness (satisfying (3) and (4)). Finally, it also provides common filters for general usage and customized filters through predefined interfaces (satisfying (5)).

*3.2. Architecture of SCondi.* SCondi has two key components: context channel and channel selector. The context channel is a transmission facility used to convey a collection of contextual data specified by the channel creator. The channel selector receives raw data from external data providers (e.g., sensors, SNS data, calendar data, email, etc.), spreading each raw context data to all the context channels that need the contextual data as their constituent, using the MQTT messaging facility as shown in Figure 1. To receive the required data, context channels should subscribe to the channel selector. When receiving the collection of the associated contextual data, the context channel delivers the collection of data to each subscribers of the channel after processing the collection of data with the filter chain through the MQTT messaging facility. In this way, the MQTT message delivery mechanism is used twice to pass the associated data to channel subscribers.

A topic with unique namespace in MQTT is allocated to each context channel. The topic manages the associated contextual data as subtopic, forming a hierarchal structure. In other words, A channel ID is assigned as a main topic while each contextual data is assigned as a subtopic separated by a "/" following the channel ID. Based on the MQTT messaging facility, SCondi provides decoupled one-to-many pub/sub through the context channel which allows any contextual data to be published once and multiple consumers to receive the collection of the needed contextual data.

In SCondi, a context is composed of a set of context primitives (CPs), where a CP is a set of related data that are used as a practical unity in applications. For instance, most elevators in the modern era have been fitted with several safety devices such as overload sensor, door sensor, fire sensor, gas sensor, cable sensor, and fault diagnosis module as shown in Figure 2. A manufacturer needs information such as equipped sensors and location of installed elevators for

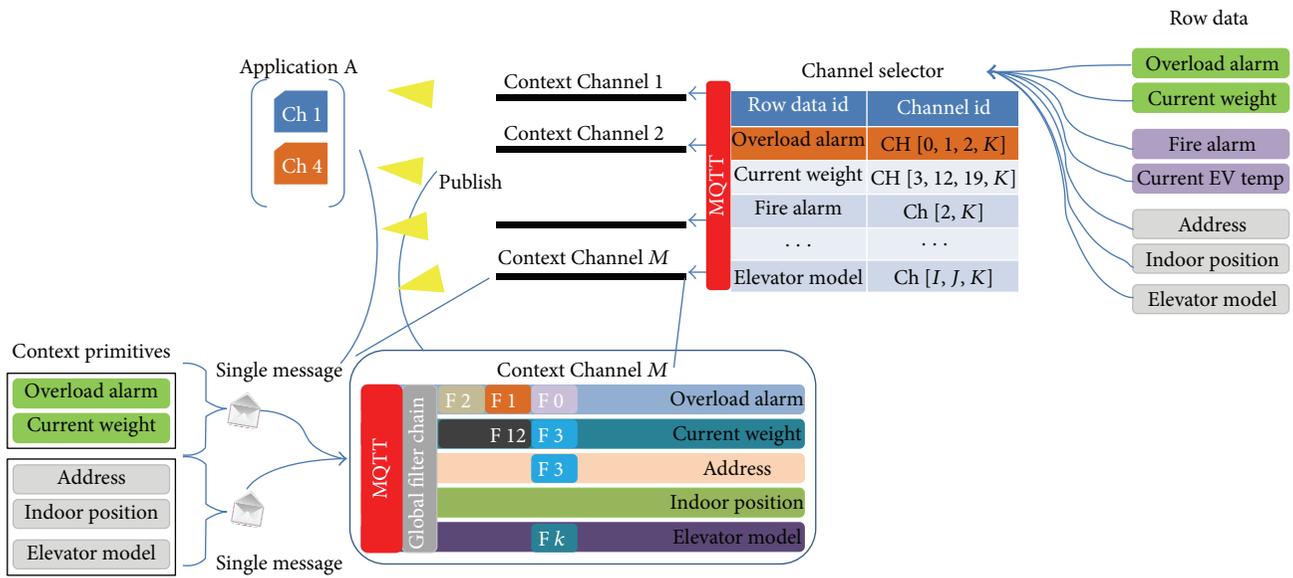


FIGURE 1: Overall structure of SCondi.

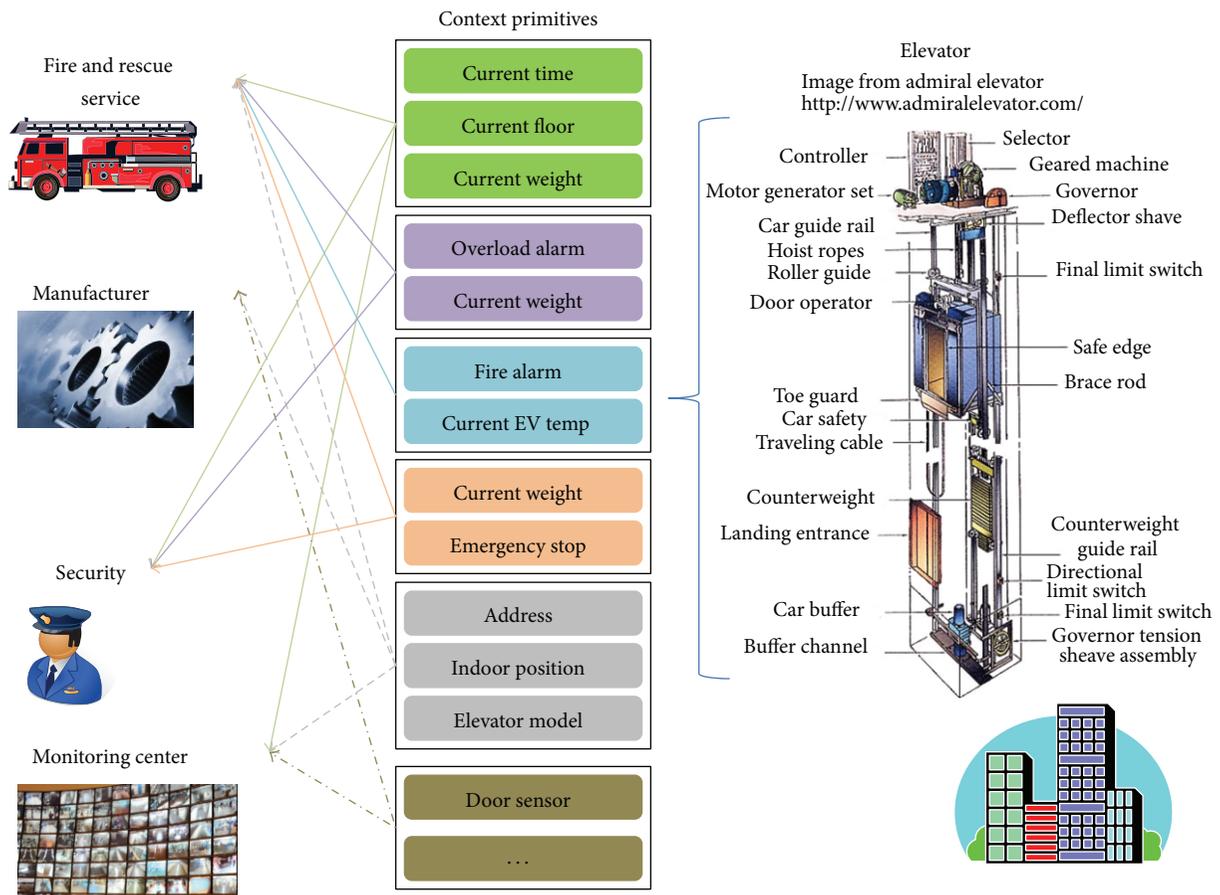


FIGURE 2: Example of context primitive.

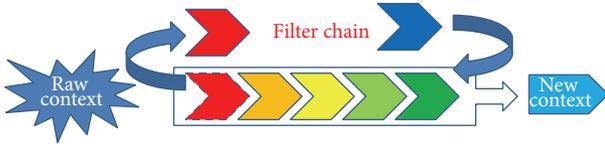


FIGURE 3: Pluggable ordered filter chain mechanism.

maintenance all together. In emergency situation, information such as floor, time, and location are required for the rescue service at the same time. As above, CP can be used effectively when information is commonly used or needed to be provided as a meaningful unit.

SCondi provides an authentication and authorization mechanism to manage access levels for context channels. To access a context channel, a subscriber should have a retrieval authority for the channel. For convenience of users, the framework supports three types of context channel according to the purpose of the application: *open*, *access-limited*, and *group channel*. An open channel allows everyone to access while an access-limited channel accepts the qualified subscriber who has the related access key. A group channel accepts the users authorized by the authentication module for the related group only.

To process contextual data with different characteristics depending on the purposes of context channels, we provide context channels with the *filter chain* mechanism. A filter chain is a collection of ordered filters. As a key feature to determine the characteristics of a context channel, the filter chain of the channel provides predefined filter interfaces to process contextual data. In addition, it allows *pluggable filter adaptation* which supports dynamic filter insertion/removal without interruption of service during run-time. For the effective run-time channel configuration, the framework uses the concept of ownership to context channels.

**3.3. Context Filter for Quality of Context.** QoC (Quality of Context) is a very important factor to be considered for the context data distribution efficiency and reliability. Traditionally QoC has focused on the quality of data only. Recently, to ensure the availability of data with the right quality, in the right place and at the right time, many related studies concentrate on complex characteristics such as data transmission time, data reliability, data accessibility, data refreshment/up-to-date, and data precision. In other words, QoC must be considered depending on the purpose of the services. To reflect these aspects, SCondi provides filter chain mechanism that allows filters to be applied for supporting QoC in context channels.

A filter changes original contextual data to qualified data according to QoC criteria. Also, as shown in Figure 3, filters can be logically combined to create more complex subscriptions patterns depending on the characteristics of context channels. In other words, filter chain mechanism helps to reduce bandwidth and to enhance scalability and to increase QoC for context distribution in the IoT environment. Our framework provides common filters that can be

TABLE 1: Filter interface.

| Interface                        | Description                                                                      |
|----------------------------------|----------------------------------------------------------------------------------|
| <code>execute(context)</code>    | Changes original contextual data to qualified data according to the filter logic |
| <code>nextFilter(context)</code> | After <code>execute()</code> , passes context to next filter in the filter chain |

TABLE 2: Filter chain interface.

| Interface                                     | Description                                                    |
|-----------------------------------------------|----------------------------------------------------------------|
| <code>addLast(filter)</code>                  | Adds the specified filter at the beginning of this chain       |
| <code>addFirst(filter)</code>                 | Adds the specified filter at the end of this chain             |
| <code>addBefore(filter, base filter)</code>   | Adds the specified filter before the base filter in this chain |
| <code>addAfter(filter, base filter)</code>    | Adds the specified filter after the base filter in this chain  |
| <code>invoke(context)</code>                  | Invokes the filters of the chain in order                      |
| <code>delete(filter)</code>                   | Deletes the specified filter in this chain                     |
| <code>replace(base filter, new filter)</code> | Replace the specified filter with new filter                   |

commonly used in context channels for QoC constraints such as `value_range`, `time`, `value_changed`, and `average filter`.

- (1) The `value_range` filter passes values only within a certain range.
- (2) The `time` filter passes values within a specific time period.
- (3) The `value_changed` filter passes values different from previous values.
- (4) The `average filter` calculates the average value with the specified condition, transmitting the calculated value.

In addition, SCondi provides predefined interfaces to create a custom filter and to manage filter chain apart from the common filters.

## 4. Implementation of SCondi

In this section, we explain the implementation of context distribution framework to support reliable delivery of context data. SCondi is implemented with Java program language, using Mosquitto as the MQTT message broker to support efficient and reliable messaging. The context channel provides a filter chain mechanism through the filter interface and the filter chain interface. The *filter interface* provides `execute()` to process contextual data through a specific filter as shown in Table 1. To pass the processed contextual data to the next filter in the filter chain, it provides `nextFilter()`. Additionally, as shown in Table 2, it provides the filter chain interface to add, delete, and replace a filter in a chain. The *filter chain interface* provides `addFirst()`, `addLast()`, `addBefore()`, and `addAfter()` to add filters at a specific position in a chain. It also gives `delete()` to remove filters and `replace()` to replace filter with a new one.

```

<Context>
 <Context Primitive1>
 <Overload Alarm>OFF</Overload Alarm>
 <Current Weight>120 KG</Current Weight>
 </Context Primitive1>
 <Context Primitive2>
 <Fire Alarm>OFF</Fire Alarm>
 <Current EV Temp>24°C</... >
 </Context Primitive 2>
 <Context Primitive 3>
 <Address>
 </Context Primitive 3>
</Context>

```

ALGORITHM 1: Context definition.

Lastly, It provides *invoke()* to execute the filters of the chain in order. In this way, the context channel can be configured to have various characteristics depending on the ordering of filters. A context is defined by a set of CPs as described in Algorithm 1.

Algorithm 2 shows an example of using a filter chain. In this case, the associated context channel executes *AverageFilter* for calculating average value of 10 recent weights of the underlying elevator. Then, *ChangedValueFilter* compares the calculated value with the previous average value. If the calculated value is not equal to the previous one, the context channel delivers the calculated value to subscribed applications.

According to the rapidly increasing number of devices in the IoT, both context channels and the channel selector can impose heavy overloads on a single MQTT message broker. Thus, SCondi uses 2 message brokers for the channel selector and context channels, respectively. A topic of the first Mosquitto message broker (for channel selection) is assigned to each source of contextual data which is provided by the external context adapter. The channel selector manages a contextual data and the associated channels through the context-channel mapping table, publishing the data to the message broker through the assigned topic. To subscribe a specific contextual data, each context channel should request a subscription of the associated topic to the message broker. After acquiring the approval from subscription authorization, the context channel can receive the interested data from the message broker by subscribing to the associated topic.

After processing the received data through its filter chain, the context channel publishes the specified set of data to the second message broker (for end user delivery). To receive the set of interested contextual data from the context channel, end user applications should subscribe to the context channel. Our framework also provides the management facility for subscription permissions on context channels. In addition, the channel provides the context-filter mapping table to manage each context data and its related filters. The channel supports 2 types of filter: *global filter* and *local filter*. Whole context data in the channel is affected by the global filter while

```

...
FilterChain =
 Channel.getFilterChain(CurrentWeight)
 FilterChain.addFirst(new AverageFilter(10));
 FilterChain.addLast(new ChangedValueFilter());
...

class AverageFilter
 implemented Filter
{

 Object execute(Object c){
 sum += c;
 if (checkCount()){
 result = sum/avgCount;
 init();
 return nextFilter.execute(result);
 }
 }

}

class ChangedValueFilter
 implemented Filter{

 Object execute(Object c){
 old = cur;
 cur = c;
 if (isChanged() == true)
 return nextFilter.execute(cur);
 ...
 }
}

```

ALGORITHM 2: Example of filter chain.

local filter is applied only to the specific context data based on the table.

## 5. Performance Analysis

Since SCondi provides higher level abstract mechanism for information delivery in the IoT environment, the qualitative effectiveness of our framework is very clear. So, we focus on the effectiveness in terms of quantity of delivered messages as illustrated in Figure 4. Whereas each data should be transmitted in an original MQTT message broker, all data in a CP can be passed as a single message through our context channel with appropriate setting. To show effectiveness of our framework, we compare the quantity of delivered messages in a formal way. For this, let us begin by defining the following terms:

$t_d$ : total number of source data;

$t_p$ : total number of CPs;

$m_{pd}$ : average number of source data in a single CP.

Then, in case that every CP is used once by application, the amount of source messages can be written as

$$t_p * m_{pd}. \quad (1)$$

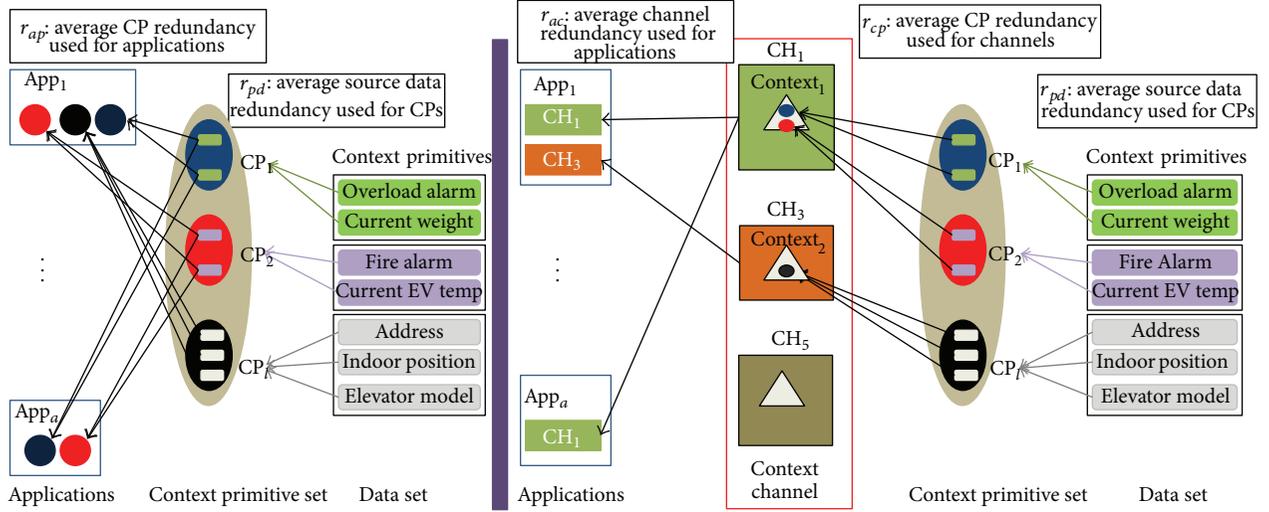


FIGURE 4: Context distribution with/without context channel.

In general, a source data can belong to multiple CPs. So we define  $r_{pd}$  denoting the average source data redundancy used for CPs:

$$r_{pd} = t_p * \frac{m_{pd}}{t_d} \quad (r_{pd} > 0). \quad (2)$$

Let  $t_a$  denote the total number of applications and  $m_{ap}$  denote average number of CPs in a single application.

For simplicity, assume that every application receives its needed data once for a certain period of time  $p$ . Then, the total number of CPs used during time interval  $p$  is

$$t_a * m_{ap}. \quad (3)$$

Since a CP is generally required by multiple applications, we define the average CP redundancy used for applications denoted by  $r_{ap}$ :

$$r_{ap} = t_a * \frac{m_{ap}}{t_p} \quad (r_{ap} > 0). \quad (4)$$

Now, we can calculate the amount of all messages (denoted by  $S_b$ ) passing through the original MQTT message broker:

$$S_b = t_a * m_{ap} * m_{pd}. \quad (5)$$

Applying (2) and (4) to (5), we have the following:

$$S_b = r_{ap} * r_{pd} * t_d. \quad (6)$$

Now, we measure the amount of all messages in our SCondi. We start by new terms  $t_c$  and  $m_{cp}$  that denote the total number of channels in SCondi and the average number of CPs used in a single channel, respectively. We assume that every context channel receives its needed CP once for a certain period of time  $p_1$ . Then, the total number of CPs passed during time interval  $p_1$  can be written as

$$t_c * m_{cp}. \quad (7)$$

Since a CP generally belongs to multiple context channels,  $r_{cp}$  which is the average CP redundancy used for channels can be defined by

$$r_{cp} = t_c * \frac{m_{cp}}{t_p} \quad (r_{cp} > 0). \quad (8)$$

We assume that every application receives its needed CP once from the related context channels for a certain period of time  $p_2$ . Let  $m_{ac}$  denote the average number of channels required in a single application. Since a context channel can belong to multiple applications, let us define  $r_{ac}$  meaning the average channel redundancy used for applications by

$$r_{ac} = t_a * \frac{m_{ac}}{t_c} \quad (r_{ac} > 0). \quad (9)$$

Recall that each data should be transmitted in an original MQTT message broker while all data in a CP can be passed as a single message from context channels with appropriate setting. Now, we can count the amount of all messages (denoted by  $S_c$ ) passing through SCondi during time period  $(p_1 + p_2)$ :

$$S_c = t_a * m_{ac} * m_{cp} + t_c * m_{cp} * m_{pd}. \quad (10)$$

Applying (2), (8), and (9) to (10), we have the following:

$$\begin{aligned} S_c &= t_c * r_{ac} * m_{cp} + t_p * r_{cp} * m_{pd} \\ &= r_{ac} * r_{cp} * t_c + r_{cp} * r_{pd} * t_d. \end{aligned} \quad (11)$$

Since the time required to transmit messages is fairly smaller than the time interval that applications require source data, we can assume  $p = p_1 + p_2$  without loss of generality. Now, we discuss the condition over which the amount of all messages passing through SCondi is less than the amount of

all messages passing through original MQTT message broker. Consider

$$S_b > S_c = r_{ap} * r_{pd} * t_d > r_{ac} * r_{cp} * t_p + r_{cp} * r_{pd} * t_d$$

(applying (6), (11)).

(12)

Note that the condition

$$r_{ap} > r_{cp} \quad (13)$$

should always satisfy to hold  $S_b > S_c$ , since  $r_{ac} * r_{cp} * t_p$  is always greater than zero. Consider

$$(r_{ap} - r_{cp}) * r_{pd} * t_d > r_{ac} * r_{cp} * t_p. \quad (14)$$

In case that  $r_{ap} > r_{cp}$ , (14) is equivalent to

$$r_{pd} * \frac{t_d}{t_p} > r_{ac} * \frac{r_{cp}}{r_{ap} - r_{cp}}. \quad (15)$$

Since  $r_{pd} = t_p * m_{pd} / t_d$  (by (2)), we finally have the following:

$$m_{pd} > \frac{r_{ac} * r_{cp}}{(r_{ap} - r_{cp})}. \quad (16)$$

To capture the meaning of conditions for  $S_b > S_c$ , using a practical data, let us calculate the amount of total delivered messages in SCondi and that of MQTT broker. We assume that  $t_d = 100000$ ,  $t_p = 80000$ ,  $t_a = 5000$ , and  $m_{pd} = 3$ . For MQTT broker case, we assume the average number of CPs in a single application is 200. In SCondi, we assume that the total number of channels is 2000, the average number of CPs in a single channel is 50 and the average number of channels in a single application is 4. Table 3 shows the calculation of the total delivered messages in both cases. As a result,  $S_b$  is 3000000 and  $S_c$  is 1300000 when  $m_{pd}$  is 3. In this case, both conditions (13) and (16) " $r_{ap}(= 12.5) > r_{cp}(= 1.25)$ " and " $m_{pd}(= 3) > r_{ac} * r_{cp} / (r_{ap} - r_{cp})(= 1.11)$ " are satisfied, resulting in  $S_c < S_b$ .

Figure 5 depicts the total amount of messages according to the value of  $m_{pd}$  under the same conditions. Considering the result, even in the aspect of the amount of total messages passing through, the concept of context channel is very useful and effective when CP is composed of two or more source data.

## 6. Conclusion

In this paper, we introduced a context distribution framework named SCondi that supports effective dissemination of context data through context channels. SCondi is based on two major components: channel selector and context channel. The channel selector sends each raw context data to each of the context channels that requires the contextual data, using the MQTT messaging facility which has been adopted by OASIS as a standard messaging facility for the IoT. The context channel provides a filter chain mechanism that supports effective extraction, tailoring, authentication,

TABLE 3: Example of delivered message comparison with practical data.

| Common conditions                                   |                                                |
|-----------------------------------------------------|------------------------------------------------|
| $t_d = 100000, t_p = 80000, t_a = 5000, m_{pd} = 3$ |                                                |
| MQTT broker case                                    | SCondi case                                    |
| Let                                                 | Let                                            |
| $m_{ap} = 200$                                      | $t_c = 2000, m_{ac} = 4, m_{cp} = 50$          |
|                                                     | $S_c = t_a * m_{ac} * m_{cp}$                  |
| $S_b = t_a * m_{ap} * m_{pd}$                       | $+ t_c * m_{cp} * m_{pd}$                      |
| $= 5000 * 200 * 3$                                  | $= 5000 * 4 * 50$                              |
| $= 3000000$                                         | $+ 2000 * 50 * 3$                              |
|                                                     | $= 1300000$                                    |
| $r_{pd} = t_p * m_{pd} / t_d$                       | $r_{cp} = t_c * m_{cp} / t_p$                  |
| $= 80000 * 3 / 100000$                              | $= 2000 * 50 / 80000$                          |
| $= 2.4$                                             | $= 1.25$                                       |
| $r_{ap} = t_a * m_{ap} / t_p$                       | $r_{ac} = t_a * m_{ac} / t_c$                  |
| $= 5000 * 200 / 80000$                              | $= 5000 * 4 / 2000$                            |
| $= 12.5$                                            | $= 10$                                         |
|                                                     | $m_{pd} > r_{ac} * r_{cp} / (r_{ap} - r_{cp})$ |
|                                                     | $> 10 * 1.25 / (12.5 - 1.25)$                  |
|                                                     | $> 1.11$                                       |

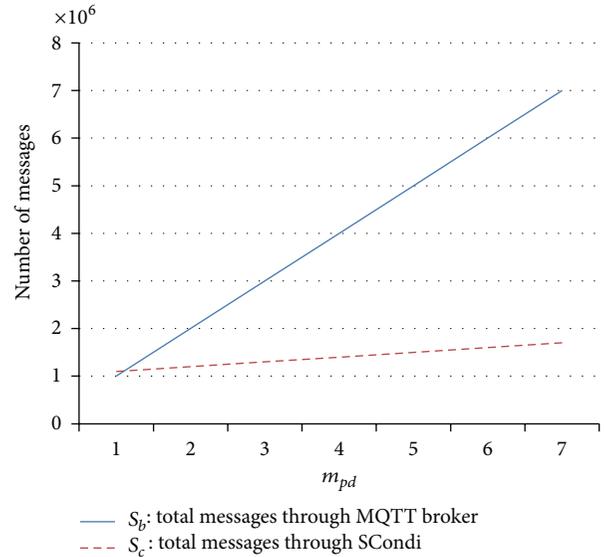


FIGURE 5:  $m_{pd}$  effect on total amount of messages.

and security of information through various types of filters. Based on the MQTT messaging facility again, when receiving the collection of the associated contextual data, the context channel delivers the collection of data to each subscriber of the channel after processing the collection of data with the filter chain. In addition, SCondi supports three types of

context channel according to the purpose of the application: open, access-limited, and group channel. We also showed that SCondi is very useful and effective both in quality and quantity of delivered messages. We believe that the novel concept of context channel and the presented framework can be very useful for context distribution in the IoT environment.

## Conflict of Interests

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

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (no. 2013R1A1A4A01004459).

## References

- [1] B. N. Schilit, N. I. Adams, and R. Want, "Context-aware computing applications," in *Proceedings of the Workshop on Mobile Computing Systems and Applications*, pp. 85–90, IEEE Computer Society, Santa Cruz, Calif, USA, December 1994.
- [2] J. Park, H. Lee, and M. Lee, "JCOOLS: a toolkit for generating context-aware applications with JCAF and DROOLS," *Journal of Systems Architecture*, vol. 59, no. 9, pp. 759–766, 2013.
- [3] D. Gallego and G. Huecas, "An empirical case of a context-aware mobile recommender system in a banking environment," in *Proceedings of the 3rd FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing (MUSIC '12)*, pp. 13–20, IEEE, Vancouver, Canada, June 2012.
- [4] S. Oh, "Using an adaptive search tree to predict user location," *Journal of Information Processing Systems*, vol. 8, no. 3, pp. 437–444, 2012.
- [5] "M2M, machine-to-machine," <http://www.m2m.com/>.
- [6] ABI Research, *More Than 30 Billion Devices Will Wirelessly Connect to the Internet of Everything in 2020*, ABI Research, London, UK, 2013.
- [7] 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.
- [8] N. Ahmed Surobhi and A. Jamalipour, "A context-aware M2M-based middleware for service selection in mobile ad-hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, 2014.
- [9] V. Cristea, C. Dobre, and F. Pop, "Context-aware environments for the internet of things," in *Internet of Things and Inter-cooperative Computational Technologies for Collective Intelligence*, vol. 460 of *Studies in Computational Intelligence*, pp. 25–49, Springer, Berlin, Germany, 2013.
- [10] "Internet of Things Strategic Research Roadmap," CERP- IoT, April 2011.
- [11] REST, <http://tools.ietf.org/html/rfc2616>.
- [12] MQTT, <http://mqtt.org/>.
- [13] V. Lampkin, W. T. Leong, L. Olivera, S. Rawat, N. Subrahmanyam, and R. Xiang, *Building Smarter Planet Solutions with MQTT and IBM WebSphere MQ Telemetry*, IBM Redbooks, New York, NY, USA, 1st edition, 2012.
- [14] XMPP, <http://xmpp.org/>.
- [15] CoAP, <https://datatracker.ietf.org/doc/draft-ietf-core-coap/>.
- [16] T. Teraoka, "Organization and exploration of heterogeneous personal data collected in daily life," *Human-Centric Computing and Information Sciences*, vol. 2, no. 1, pp. 1–15, 2012.
- [17] Mosquitto, <http://mosquitto.org/>.
- [18] MQTT Servers/Brokers, <http://mqtt.org/wiki/doku.php/brokers>.
- [19] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing (HUC '99)*, vol. 1707 of *Lecture Notes in Computer Science*, 1999, pp. 304–307.
- [20] P. Bellavista, A. Corradi, M. Fanelli, and L. Foschini, "A survey of context data distribution for mobile ubiquitous systems," *ACM Computing Surveys*, vol. 44, no. 4, article 24, 2012.

## Research Article

# A Prediction System Using a P2P Overlay Network for a Bus Arrival System

Ssu-Hsuan Lu<sup>1</sup> and Yu-Wei Chan<sup>2</sup>

<sup>1</sup> Department of Computer Science, National Tsing Hua University, Hsinchu 300, Taiwan

<sup>2</sup> Department of Information Management, Chung Chou University of Science and Technology, Changhua 510, Taiwan

Correspondence should be addressed to Yu-Wei Chan; [ywchan@dragon.ccut.edu.tw](mailto:ywchan@dragon.ccut.edu.tw)

Received 20 January 2014; Accepted 15 July 2014; Published 24 August 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 S.-H. Lu and Y.-W. Chan. 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.

Along with the evolution of times and the surge of metropolitan populations, government agencies often promote the construction of public transport. Unlike rail transportation or rapid transit systems, it is often difficult to estimate the vehicle arrival times at each station in a bus transportation system due to metropolitan transportation congestion. Traffic status is often monitored using wireless sensor networks (WSNs). However, WSNs are always separated from one another spatially. Recent studies have considered the connection of multiple sensor networks. This study considers a combination view of peer-to-peer (P2P) overlay networks and WSN architecture to predict bus arrival times. Each bus station, which is also a P2P overlay peer, is connected in a P2P overlay network. A sensor installed in each bus can receive data via peers to obtain the moving speed of a bus. Then, each peer can exchange its data to predict bus arrival times at bus stations. This method can considerably increase the accuracy with which bus arrival times can be predicted and can provide traffic status with high precision. Furthermore, these data can also be used to plan new bus routes according to the information gathered.

## 1. Introduction

Over the last decade, medical advances and rapid economic development have led to a substantial increase in population, which in turn causes increased traffic. Therefore, governments have attempted to reduce the number of residents who drive cars and the corresponding environmental pollution by developing public transportation. Such measures also bring in additional urban tourism resources. One of the factors that affect whether people are willing to take public transportation is the accuracy of the arrival time when using this mode of transportation.

Residents in many cities are not familiar with public transportation services and may not know how to take public transportation. This lack of knowledge arises from the fact that people are used to driving autonomously. The most serious weakness of public transportation is that it is not systematic, which makes it difficult to shorten the time required to take public transportation. The lack of systematic

and interpretable information, such as the positions of bus stations and bus routes, introduces considerable uncertainty when taking public transportation. The provision of information regarding public transportation is dependent on relevant units, particularly for individuals who are not proficient at determining directions and using maps. If clear and comprehensible information can be provided to individuals, it allows them to easily predict their use of time and plan their schedules, thus making them more willing to take public transportation to reach their desired destinations.

In public transportation, the accuracy of the arrival time of buses is the most difficult information to predict. The arrival times of rail lines can be more precisely predicted because the routes of public rail transportation are fixed and not easily disturbed during locomotion. In contrast, buses drive on the same infrastructure as general cars, and thus the prediction accuracy of the arrival time is easily affected by traffic conditions. This issue has become a major obstacle to develop bus arrival time prediction systems.

Arrival time prediction systems for public transportation are often developed with the use of modern wireless telecommunications, including wireless sensor networks (WSNs) [1–6]. A WSN is composed of a large number of inexpensive microsensor nodes deployed in the monitoring area [7, 8] and communicates via wireless communication [9]. A WSN is easy to deploy, programmable, and dynamically reconfigurable [10, 11].

WSNs are also suitable for applications that require network provision system environments that are fast, easy, or impossible to preestablish; WSNs can gradually replace the traditional sensor application systems that are monitored using artificial methods. WSNs can efficiently and automatically retrieve data using wireless network transmission.

Existing bus arrival time prediction systems transmit the data collected to centralized servers. However, this type of system has considerable overhead in bus systems in fairly developed areas. Because many buses drive on roads simultaneously, a significant amount of data is sent to the server in certain periods. The server must analyze and calculate these data to predict the arrival time of each bus. These data must be processed and sent in real time, so they cannot be delayed by any system overhead; otherwise, the prediction accuracy will be affected.

This paper proposes a method of combining a peer-to-peer (P2P) overlay network and WSN to develop a bus arrival time prediction system. A P2P overlay network is added to traditional prediction systems to allow for real-time data. Each bus is installed with a sensor, and each bus stop can receive data sent from sensors. Due to the distance limitation of sensors, the sensors on buses and the data-receiving device of a bus station form a single WSN environment. All bus stations and station termini are connected to form a P2P overlay network, which is used to transmit real-time bus information and predict bus arrival times. Through the WSN technology, bus stations retrieve data from buses and transmit these data to subsequent bus stations to estimate the bus arrival times. This approach can be a powerful tool for monitoring and predicting traffic conditions.

The advantage of using P2P overlay networks to connect bus stations is that each bus station can act as a client and server to exchange information for nearby buses. Through this method, the data collected from buses do not need to be transmitted from bus stations to a centralized server and could be evaluated such that bus stations can predict the arrival time. This paper adopts an Arrangement Graph-based Overlay network (AGO) [12, 13] as our P2P overlay network because this system performs well in transmitting messages. By combining P2P overlay networks and WSNs, this study considers a novel approach to predict bus arrival times and presents some initial results of this endeavor.

Some experiments were performed to demonstrate the performance of our bus arrival time prediction system compared to existing prediction systems; the experimental results revealed that our bus arrival time prediction system can make more accurate predictions. Although there are several factors that can affect the accuracy of our bus arrival time prediction system, such as the time of day and number of passengers, our prediction system is more convenient and provides more

information for passengers than existing prediction systems, particularly paper timetables. The number of messages transmitted to the centralized server and the system overhead of the centralized service are both reduced considerably.

The remainder of this paper is organized as follows. Section 2 presents some related work on WSNs, P2P overlay networks, and AGO systems. Section 3 describes the proposed bus arrival time prediction system, and some experimental results are presented in Section 4. Finally, the conclusions of this study and potential avenues for future work are discussed in Section 5.

## 2. Related Work

In this section, some studies related to our proposed system are introduced. Our bus arrival time prediction system is developed by using WSNs and P2P overlay networks, and thus some properties related to these technologies are introduced.

*2.1. Wireless Sensor Networks.* Due to the rapid advancements in microfabrication, communication, and embedded processing, small, sophisticated electronic devices can be embedded with sensing, computing, and communicating functions. Therefore, WSNs have become a popular research topic in computer science. WSNs mainly include sensing, communicating, and computing aspects (i.e., hardware, software, and algorithms, resp.). A WSN is a network system composed of several wireless data receivers and sensors, and communication between these components is wireless communication. To achieve large-scale deployment, WSN devices should be inexpensive, small, and easy to deploy and should have low power consumption [14, 15]. These devices also need to have sensing, programmability, and dynamic reconfiguration capabilities because sensors rely on the power of batteries to supply the energy necessary for operations and radio transmission distance. Sensor nodes transmit and receive data via wireless technology, and sensor networks are largely used for short-distance data transmission to reduce power consumption.

The development of WSNs initially originated in military applications, such as battlefield monitoring by the University of California, Berkeley (UC, Berkeley), for a research project called Smart Dust [16] funded by the United States Defense Research Projects Agency (DARPA). Many manufacturers have followed the direction of research by combining IEEE 802.15.4 low-rate wireless personal area networks (LR-WPANs) and ZigBee [17, 18].

Many standard applications use common wireless communication technologies, including automated home devices, online shopping [19], environment safety and control, and personal health care [20, 21].

*2.2. Peer-to-Peer Overlay Network.* In the recent past, an information system was typically a single server that handled all requests from clients and all responses to them. Clients had to first talk to the server to establish communication channels and then sent requests to the server for processing. If there was any information that needed to be communicated

between clients, all information also needed to be sent to the server. This scheme is referred to as the client-server architecture. However, the service performance of the overall system is limited by the computing power of the server and the bandwidth of external networks, which can easily reduce the performance of system services. All clients are connected to a single server, and thus if the administrator wants to improve the system performance, the only option is to upgrade the central server and increase the bandwidth of the external network.

Therefore, P2P overlay networks emerged. The P2P overlay network is an abstract network that ignores physical network connections. Each peer on the network is treated as an individual peer and assumes that they can freely interconnect. These individual peers are connected using some specific topology to build an overlay network, which may be composed of several physical network connections. Some desired effects can be achieved through an overlay network of interconnected peers. All peers in the overlay network can both act as a client to obtain data from other peers and play the role of a server to provide data to other peers. One of the important results is to allow all peers to pool their resources, including network bandwidth, storage space, and computing power, which can increase the flexibility of the network considerably [22].

A P2P overlay network can be structured or unstructured. A structured P2P overlay network, such as Chord [23, 24], Pastry [25], and Kademlia [26], often uses a distributed hash table (DHT) to determine connections. In contrast, an unstructured P2P overlay network, such as Gnutella [27], does not have this connection relation. Thus, the unstructured P2P overlay network often uses flooding and time-to-live (TTL) to make queries on the overlay network [28, 29].

**2.3. Arrangement Graph-Based Overlay.** The AGO is a P2P overlay network that was developed based on arrangement graph [30]. The arrangement graph is an undirected graph. Two parameters,  $n$  and  $k$ , are used to define the arrangement graph. Each peer in the arrangement graph has a unique peer ID used for identification. The parameter  $k$  is the number of digits in the peer IDs,  $n$  is the range of each digit, and  $1 \leq k \leq n - 1$ . Some properties of the arrangement graph are as follows: there are  $n!/(n - k)!$  peers, the degree of each peer is  $k(n - k)$ , and the diameter of an arrangement graph is  $\lfloor 3k/2 \rfloor$ . Moreover, the peer in the arrangement graph has only one digit that is different from its neighbors.

Because the AGO was developed based on the arrangement graph, the AGO inherited some arrangement graph properties. Peers in the AGO establish their neighbor tables according to properties of the arrangement graph and the main information of their  $k(n - k)$  neighbors. There are three main functions in AGO: *joining*, *departing*, and *routing*. Peers initiate the joining action to join the AGO through the bootstrap peer. There is a *waiting peer pool* in the bootstrap peer to temporarily record information of peers that already exist in the AGO. These records are provided to the new peers joining the AGO. Moreover, the peer record in the waiting peer pool is replaced when its neighbor table is full.

When the peer in AGO tries to depart, it sends announcement messages to its neighbors to maintain the AGO's accuracy. Furthermore, peers can discover information on other peers using the routing process. The AGO utilizes one of the properties of the arrangement graph, the digit difference, to perform routing actions. A replica mechanism is also used to increase routing performance. Furthermore, the AGO can also self-extend or self-shrink the scale of the AGO by adjusting the value of the parameters according to the number of peers.

### 3. Bus Arrival Time Prediction System

This section introduces the proposed method, which is a two-layer structure. One layer is a P2P overlay network that is used to transmit data between bus stations. The other layer is a WSN used by bus stations to retrieve data from buses. This structure allows bus arrival times to be estimated more efficiently.

**3.1. System Architecture.** WSNs have a wide range of applications, including public transportation. For this study, devices were installed on bus stations to retrieve data, and sensors were installed on buses to supply traffic conditions. When buses drive into the wireless coverage of bus stations, bus stations collect data from buses and transmit these data to neighboring bus stations. These data can provide real-time transport information, such as time and location, and can be used to estimate the expected arrival time for neighboring bus stations. In addition, some real-time transport information, such as traffic conditions or emergencies, can also be transmitted to other places to improve traffic safety and accident-handling efficiency.

This study utilizes the functional characteristics of WSNs and P2P overlay networks for urban public transportation to establish a multifunctional intelligent information system. A WSN is used to establish a framework for a multifunctional information platform that provides electronic toll collection, traffic monitoring, traffic statistics, and traffic and emergency notification systems.

The proposed system is composed of two parts, as shown in Figure 1. The top layer is a P2P overlay network, which is formed using an AGO system. This network is responsible for delivering the data collected from sensors to other bus stations. The bottom layer consists of WSNs composed of bus stations and buses. Bus stations receive data from sensors to judge the conditions of moving buses. There are three key components concerning actual practices: the technologies for combining the P2P overlay network and WSN, the designs for efficiently transmitting and predicting messages and user-friendly interfaces, and the indicators of performance evaluation. These components make the system more modular and can allow for cost savings and increased revenue. These components were also our goals during development of the proposed system.

**3.2. Method.** In Figure 1, the upper layer is a P2P overlay network, which is built using an AGO system. The AGO only needs to be modified with a small mechanism to apply our

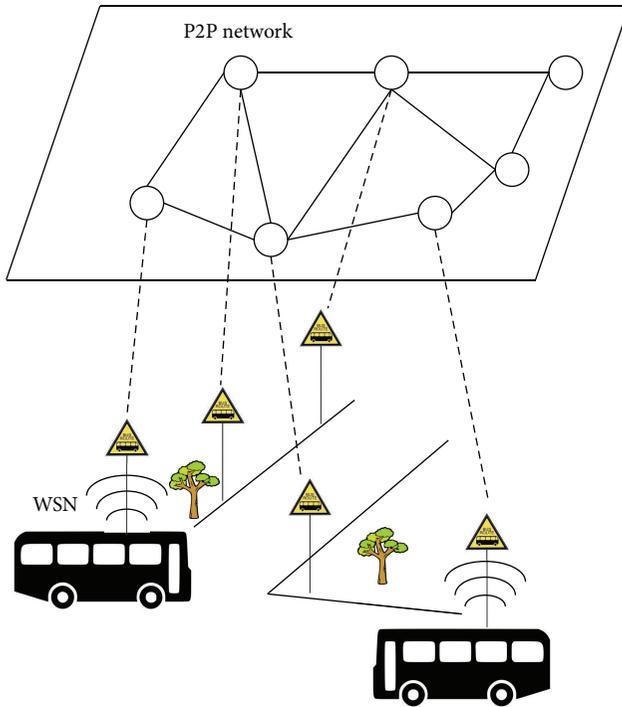


FIGURE 1: Illustration of the system platform.

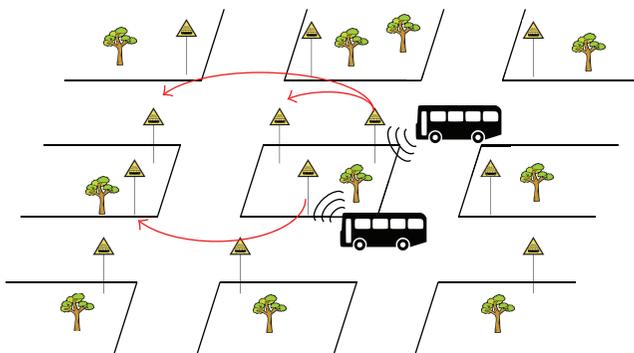


FIGURE 2: The diagram of our method.

bus arrival time prediction system. The GPS location of each bus station must be recorded to establish the P2P overlay network according to their locations. Furthermore, a user interface is designed so that users can determine when buses will arrive in the network, as shown on information boards at bus stations. Data-retrieving devices on bus stations act as peers in the P2P overlay network and are responsible for receiving information sent from sensors on buses.

When the prediction system starts every morning, bus stations start to join the P2P overlay network. The stations link to other neighboring bus stations. Then, when a bus station detects a bus driving in its coverage, the bus station starts to collect data sent from the bus, such as the speed or location of the bus. After the bus station collects these data,

it analyzes these data and sends them to its neighboring bus stations to predict bus arrival times for other bus stations. The bus station that receives data from other bus stations analyzes the received data and provides the predicted arrival time on information boards. The bus station also records the actual time in which the bus arrives. At night, when buses are out of service, bus stations upload all of the data for the day to a centralized server for storage and analysis. System operators can comprehensively analyze these data to correct the prediction system and make the system more accurate. Furthermore, the bus department can use the data to adjust the bus headway.

A diagram of our method is shown in Figure 2. Bus stations collect data from sensors on buses and send these data to other neighboring bus stations. Each bus station is connected to some other bus stations near it, when it joins the P2P overlay network according to its estimated location. Data transmitted between bus stations utilize connections in the AGO. When the next bus station receives data from the former bus stations, it can calculate the probable arrival time of the bus according to the distance, the speed of the bus, and average speed at that location. Therefore, whether a bus arrives or leaves a bus station, the bus station is required to send messages to the next bus station to enable more accurate prediction.

Bus stations are connected with those near them to prevent two adjacent bus stations from being far in the P2P overlay network. Furthermore, the main purpose of the P2P overlay network is to enable bus stations to directly exchange data with other bus stations without any mediating servers. This ability can decrease the system overhead of the centralized server considerably.

## 4. Experimental Results

In this section, simulation results are presented. The authors performed the following simulations according to the above methods. From these simulation results, the performance of our prediction system can be assessed. The data in simulations are produced according to the assumption in the paper, and the data is used for both the traditional system and our prediction system.

*4.1. Accuracy of the Arrival Time Prediction.* The accuracy of a bus arrival time prediction system is known from the status of passengers' usage. Passengers care about the accuracy of prediction systems, and the accuracy affects passengers' decision to use the prediction system. Passengers will use a prediction system if they consider it to be trustworthy, and the bus department gains revenue from their use of the system. Therefore, one of the aims of the simulations is to determine the accuracy of our bus arrival time prediction system.

Simulations were performed to simulate an 18 h experiment. Buses are assumed to begin service from 6:00 and end service at 24:00. Because our government regulates the speed to a maximum of 40 km/h, the speeds of buses are assumed to range from 20 to 40 km/h. Furthermore, the driving speeds of buses are also affected by traffic conditions. During peak

hours, the driving speeds will be lower than non-peak-hour driving speeds.

Furthermore, the duration of traffic lights ranges from 30 to 120 s. In particular, for those crossroads that have heavy traffic during peak hours, the duration of red traffic lights at these crossroads may require 120 s. For those crossroads that do not have heavy traffic or are not being used during peak hours, the time of red lights can be shorter than 120 s. However, the duration of the red phase of traffic lights should be at least 30 s.

Another factor that can affect the accuracy of the prediction time is the number of passengers. Buses spend more time at bus stations if there are more passengers. Because it will cost time for passengers to board and alight from buses, each bus will stop at the bus station for a different period of time. During peak hours, there will be more passengers, and thus the waiting time is longer than usual. Moreover, some bus stations also require more time because they are in hot spots. Thus, there will also be more passengers utilizing buses. However, this factor is not considered in our bus arrival time prediction system because one of the methods in our system has already avoided being affected by this factor. Bus stations record the bus arrival and departure times. When bus stations transmit data to other bus stations, other bus stations receive data that already consider this factor. Therefore, in our prediction system, the time passengers take to board and alight from buses does not need to be assumed.

Figure 3 presents the accuracy of our bus arrival time prediction system as a percentage. The  $x$ -axis presents the time of day that buses serve, which ranges from 6:00 to 24:00. The  $y$ -axis presents the percentage accuracy of our system. A higher percentage indicates a more accurate system.

The accuracy of our bus arrival time prediction system is over 76%. The accuracy is affected during peak hours, such as 9:00 and 18:00. From 6:00 to 9:00, people start to go to work, and traffic conditions are disturbed. Similarly, at approximately 18:00, people leave work to go home; at this time, traffic conditions are complex and difficult to predict. Therefore, predicting bus arrival times is more difficult in these two time periods. However, our prediction system can still achieve an accuracy over 76% in these periods, illustrating that our bus arrival time prediction system can consider the complex factors that affect traffic conditions.

Overall, the accuracy of our bus arrival time prediction system is very good. The accuracy reaches 76% in peak hours and 85% to 90% in nonpeak hours because our prediction system exchanges data on buses directly between bus stations. Then, bus stations can calculate and analyze the bus arrival times according to these data and traffic conditions.

**4.2. Amount of Messages Transmitted to Server.** In traditional prediction systems, the data on buses and traffic conditions are sent to a centralized server at any time. This action produces a large number of messages to transmit those data and consumes a large bandwidth. The centralized server must receive many messages and analyze them, and hence the centralized server experiences heavy loading.

However, in our bus arrival time prediction system, the data on buses and traffic conditions are analyzed and sent

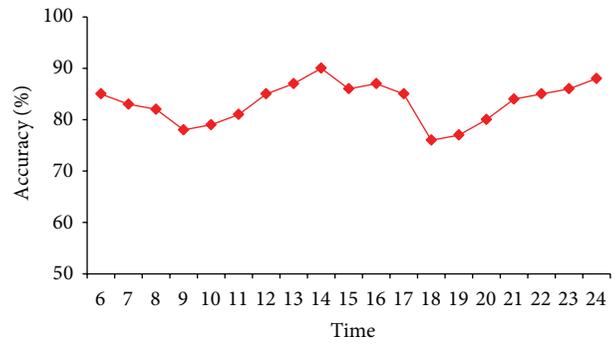


FIGURE 3: Accuracy of our prediction system (%).

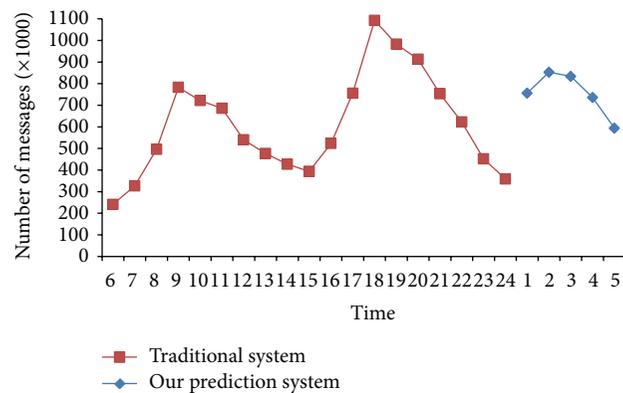


FIGURE 4: Number of messages transmitted to the centralized server.

at night when buses are out of service, and the arrival time is predicted by bus stations according to real-time data. The centralized server in our prediction system is not responsible for real-time prediction. Therefore, the arrival time prediction is not affected by the loading of the centralized server. Figure 4 presents the number of messages sent to the centralized server.

Figure 4 illustrates that traditional systems transmit data collected from buses at any time when buses are in operation. Therefore, many messages are transmitted to the centralized server simultaneously. However, because the centralized server also needs to calculate the predicted arrival time of buses, there is substantial system overhead in the centralized server. Similarly, the performance of the prediction arrival time is affected by the system overhead of the centralized server.

Our system does not transmit data collected during the day, and thus bus stations need to be responsible for predicting arrival time. Collected data are transmitted to the centralized server at night because the centralized server is only responsible for storing these data and comprehensively analyzing them for the bus department's reference. In this manner, the number of messages transmitted to the centralized server is decreased considerably, the system overhead of the centralized server is also decreased, and bus stations can calculate the arrival time of buses in real time.

## 5. Conclusions and Future Work

Existing arrival time prediction systems do not provide passengers with information about whether their buses will depart or arrive soon because of their low accuracy and limited deployment. For passengers, the arrival time of buses is important because passengers can decide whether they should wait for the next bus or choose other means of public transportation.

This study develops a bus arrival time prediction system that combines a P2P overlay network and WSN. The data obtained at bus stations and from sensors for buses are used to accurately predict arrival time. Data collected from each bus are logged into the system and used to optimize the courses and frequency of buses. Bus routes and schedules can be optimized to meet the actual needs of passengers according to data collected from bus stations. The proposed system can reduce costs and increase revenue for bus departments as well as improving passenger satisfaction.

The experimental results demonstrate that our bus arrival time prediction system can achieve high accuracy using real-time exchange of data and traffic conditions between bus stations. Furthermore, the experimental results demonstrate that our prediction system can decrease the system overhead of the centralized server considerably. The arrival time was predicted by bus stations according to the real-time data they received.

With the advance of microfabrication technology and the development of wireless transmission technology, WSNs have been used in a wide range of applications. WSNs can also be used in electronic toll collection, traffic, and road information applications to improve safety, convenience, and efficiency. The investment trends of advanced countries indicate that telematics will be an increasingly used wireless technology.

## Conflict of Interests

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

## References

- [1] B. Hamner, "Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow," in *Proceedings of the 10th IEEE International Conference on Data Mining Workshops (ICDMW '10)*, pp. 1357–1359, Sydney, Australia, December 2010.
- [2] F. Li, Y. Yu, H. Lin, and W. Min, "Public bus arrival time prediction based on traffic information management system," in *Proceedings of the IEEE International Conference on Service Operations, Logistics and Informatics (SOLI '11)*, pp. 336–341, Beijing, China, July 2011.
- [3] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transportation Research C: Emerging Technologies*, vol. 19, no. 4, pp. 606–616, 2011.
- [4] J. Song, "The application of WSN technology in the space location system," in *Proceedings of the 4th International Congress on Image and Signal Processing (CISP '11)*, vol. 2, pp. 633–636, Shanghai, China, October 2011.
- [5] B. Son, H. J. Jin, C. H. Shin, and S. K. Lee, "Bus arrival time prediction method for ITS application," in *Proceedings of 8th International Conference of Knowledge-Based Intelligent Information and Engineering Systems (KES '04)*, pp. 88–94, 2004.
- [6] R. Sumathi and M. G. Srinivas, "A survey of QoS based routing protocols for wireless sensor networks," *Journal of Information Processing Systems*, vol. 8, no. 4, pp. 589–602, 2012.
- [7] H.-Y. Jeong, "The remote management of operational information for manufacturing systems," *Journal of Convergence*, vol. 3, no. 2, pp. 45–50, 2012.
- [8] E.-H. Song, H.-W. Kim, and Y.-S. Jeong, "Visual monitoring system of multi-hosts behavior for trustworthiness with mobile cloud," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 347–358, 2012.
- [9] WSN, 2013, [http://en.wikipedia.org/wiki/Wireless\\_sensor\\_network](http://en.wikipedia.org/wiki/Wireless_sensor_network).
- [10] Q.-L. Liu and D.-H. Oh, "Performance evaluation of multi-hop communication based on a mobile multi-robot system in a subterranean laneway," *Journal of Information Processing Systems*, vol. 8, no. 3, pp. 471–482, 2012.
- [11] K. Sohrawy, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*, Wiley, 2007.
- [12] S.-H. Lu, K.-C. Lai, K.-C. Li, and Y.-C. Chung, "Design and analysis of arrangement graph-based overlay systems for information sharing," in *Proceedings of the 3rd IEEE International Workshop on Management of Emerging Networks and Services (IEEE MENS '11)*, pp. 668–672, Houston, Tex, USA, December 2011.
- [13] S.-H. Lu, K.-C. Li, K.-C. Lai, and Y.-C. Chung, "Arrangement graph-based overlay with replica mechanism for file sharing," in *Proceedings of the 12th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN '12)*, pp. 192–200, San Marcos, Tex, USA, December 2012.
- [14] G. Gutiérrez, B. Mejías, P. van Roy, D. Velasco, and J. Torres, "WSN and P2P: a self-managing marriage," in *Proceedings of the 2nd IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW '08)*, pp. 198–201, Venice, Italy, October 2008.
- [15] B. Singh and D. K. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," in *Human-Centric Computing and Information Sciences*, vol. 2, 2012.
- [16] B. Warneke, M. Last, B. Liebowitz, and K. Pister, "Smart dust: communicating with a cubic-millimeter computer," *Computer*, vol. 34, no. 1, pp. 44–51, 2001.
- [17] IEEE .802.15.4, [http://en.wikipedia.org/wiki/IEEE\\_802.15.4](http://en.wikipedia.org/wiki/IEEE_802.15.4).
- [18] Zigbee, 2013, <http://en.wikipedia.org/wiki/ZigBee>.
- [19] T.-T. Truong, M.-T. Tran, and A.-D. Duong, "Improvement of the more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on ECC," in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA '12)*, pp. 698–703, Fukuoka, Japan, March 2012.
- [20] J. Kee-Yin Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.
- [21] T. Teraoka, "Organization and exploration of heterogeneous personal data collected in daily life," in *Human-Centric Computing and Information Sciences*, vol. 2, 2012.

- [22] T. Ohkawara, A. Aikebaier, T. Enokido, and M. Takizawa, "Quorums-based replication of multimedia objects in distributed systems," *Human-Centric Computing and Information Sciences*, vol. 2, article 11, 2012.
- [23] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: a scalable peer-to-peer lookup service for internet applications," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '01)*, pp. 149–160, San Diego, Calif, USA, August 2001.
- [24] I. Stoica, R. Morris, D. Liben-Nowell et al., "Chord: a scalable peer-to-peer lookup protocol for Internet applications," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 17–32, 2003.
- [25] A. Rowstron and P. Druschel, "Pastry: scalable, distributed object location and routing for large-scale peer-to-peer systems," in *Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg (Middleware '01)*, vol. 2218 of *Lecture Notes in Computer Science*, pp. 329–350, Heidelberg, Germany, 2001.
- [26] P. Maymounkov and D. Mazieres, "Kademlia: a peer-to-peer information system based on the XOR metric," in *Proceedings of the 1st International Workshop on Peer-to-Peer Systems*, pp. 53–65, Cambridge, Mass, USA, March 2002.
- [27] E. Keong Lua, J. Crowcroft, P. Marcelo, R. Sharma, and L. Steven, "A survey and comparison of peer-to-peer overlay network schemes," *Journal of IEEE Communications Surveys & Tutorials*, vol. 7, no. 2, pp. 72–93, 2005.
- [28] S. Jiang, L. Guo, and X. Zhang, "LightFlood: an efficient flooding scheme for file search in unstructured peer-to-peer systems," in *Proceedings of the International Conference on Parallel Processing (ICPP '03)*, pp. 627–635, Kaohsiung, Taiwan, October 2003.
- [29] S. Jiang, L. Guo, and Z. Xiaodong, "LightFlood: minimizing redundant messages and maximizing scope of peer-to-peer search," *IEEE Transactions on Parallel and Distributed Systems*, vol. 9, no. 5, pp. 601–614, 2008.
- [30] K. Day and A. Tripathi, "Arrangement graphs: a class of generalized star graphs," *Information Processing Letters*, vol. 42, no. 5, pp. 235–241, 1992.

## Research Article

# Secure Collaborative Key Management for Dynamic Groups in Mobile Networks

Sukin Kang, Cheongmin Ji, and Manpyo Hong

Department of Computer Engineering, Ajou University, Suwon 443-749, Republic of Korea

Correspondence should be addressed to Manpyo Hong; [mphong@ajou.ac.kr](mailto:mphong@ajou.ac.kr)

Received 29 March 2014; Accepted 31 July 2014; Published 21 August 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Sukin 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.

Mobile networks are composed of heterogeneous mobile devices with peer-to-peer wireless communication. Their dynamic and self-organizing natures pose security challenge. We consider secure group key management for peer dynamic groups in mobile wireless networks. Many group based applications have achieved remarkable growth along with increasing use of multicast based services. The key sharing among the group members is an important issue for secure group communication because the communication for many participants implies that the likelihood of illegal overhearing increases. We propose a group key sharing scheme and efficient rekeying methods for frequent membership changes from network dynamics. The proposed method enables the group members to simply establish a group key and provide high flexibility for dynamic group changes such as member join or leave and group merging or partition. We conduct mathematical evaluation with other group key management protocols and finally prove its security by demonstrating group key secrecy, backward and forward secrecy, key independence, and implicit key authentication under the decisional Diffie-Hellman (DDH) assumption.

## 1. Introduction

Advances in wireless communications and mobile devices have made various types of mobile networks such as mobile ad hoc networks (MANETs), wireless mobile sensor networks (WMSNs), and Internet of things (IoT). In mobile networks, heterogeneous devices such as smartphones, laptops, and smart sensors perform peer-to-peer (machine-to-machine) communications without depending on any fixed infrastructure. Mobile networks have features distinct from conventional networks. First, network topology changes dynamically due to the mobility of nodes, which causes frequent switching of network connection state. Additionally, many applications in mobile networks support one-to-many (multicast) communication, where common data are transferred to multiple destinations from a source, for instance, military communication (battlefield), health care system, industrial monitoring, on-line conferencing, collaborative workspace, and disaster management. They build a collaborative group of

entities, called group members, which participate in multicast group communications as a group member and manage group membership changed by node mobility.

Group communication over wireless networks is susceptible to illegal overhearing such as packet sniffing. When a group deals with sensitive information, secure group communication must be achieved by sharing a common secret key—*group key* for confidentiality of group messages with data encryption. In other words, it is essential to decide how to share a key among group members and how to update the group key for group membership change [1–3]. A typical approach is based on centralized key distribution with a trusted third party (TTP) [4–8]. It provides scalable group key management for large groups using symmetric encryption such as advanced encryption standard (AES) and hierarchical logical key tree. However, it fairly depends on a constantly accessible TTP. This requirement is not suitable for mobile networks with peer-to-peer communication. To apply a symmetric key based approach without a TTP, a node

should establish secure connection for sharing a pairwise key with all other mobile nodes in a group. It requires much communication and depends on another key sharing scheme [9]. Diffie-Hellman (DH) key exchange [10] is a protocol to establish a common key based on asymmetric keys without any TTP. It allows two parties to share a key using their secrets over an insecure channel. To extend DH into group setting, group key agreement (GKA) protocols have been developed [11–16]. In the protocols, also known as contributory key agreement, all group members contribute to generation of a common key. While providing dynamic group key management, they require considerable messages or operations to establish and update group keys. An approach for reducing computation cost deploys tree structure to handle key management. Tree-based group key protocols [15–18] need to support management of tree structure and require ordered message delivery for calculation from leaves to the root of the tree.

In this paper, we investigate secure group key distribution and management for collaborative groups with high group flexibility. We propose a DH-based group key management protocol and show security proof of the proposed scheme and mathematical evaluation with other GKA protocols.

The remainder of the paper is organized as follows. In Section 2, we address related works. Section 3 explains our group key management scheme with group membership events and security requirements. Section 4 describes performance analysis and Section 5 shows security proof for the proposed key management. We conclude the paper in Section 6.

## 2. Related Work

Over the past few decades, a considerable number of studies have been conducted on group key establishment and management. A typical approach is centralized key distribution based on constantly accessible TTP and pairwise keys [4–8]. These studies showed apparent efficiency for large groups such as wireless sensor network (WSN). Since, however, a mobile network is comprised of peer-to-peer communications with dynamic mobility and without a TTP, it is difficult to provide scalable group key management on arbitrary group setting [15].

We focus on DH based group key management, known as group key agreement (GKA), in which a common key is generated by all group members' equal contributions. DH protocol allows two parties to share a key using their secrets over an insecure channel [10]. The key computation of DH uses the multiplicative group of integer modulo  $p$ , where  $p$  is a large prime number. Each party chooses a random number  $x_i$  in  $\mathbb{Z}_p$  and computes  $g^{x_i} \bmod p$ , where  $g$  is a primitive root (generator)  $\bmod p$ . They exchange the computed values,  $g^{x_1} \bmod p$  and  $g^{x_2} \bmod p$ , and agree on the common key:

$$K = (g^{x_1})^{x_2} \bmod p = (g^{x_2})^{x_1} \bmod p. \quad (1)$$

For extending it to group setting, Burmester and Desmedt (BD) proposed a conference key exchange system [11]

depending on a broadcast manner. When the number of group members is  $n$ , the group key (GK) of BD becomes

$$\text{GK} = g^{x_1 x_2 + x_2 x_3 + \dots + x_{n-1} x_n} \bmod p. \quad (2)$$

As BD system requires large communication messages, Steiner et al. proposed group key agreement protocols called group Diffie-Hellman (GDH) [12, 13]. In GDH,

$$\text{GK} = g^{x_1 x_2 \dots x_{n-1} x_n} \bmod p. \quad (3)$$

They showed not only that DH can be extended efficiently to group setting, but also that their protocol can deal efficiently with group membership change. They presented three distinct group key agreements GDH.1, GDH.2, and GDH.3, which later was advanced as a protocol suite known as CLIQUES [13]. In GDH. $x$ , group members can individually or massively join and leave; CLIQUES also considers group integration and group division. A variant of GDH protocol is a centralized key distribution (CKD) scheme. In CKD, a controller distributes the group key to every member using pairwise temporal keys between the controller and each of the members, which is computed using DH fashion.

As group dynamics have become an important issue, some studies have adopted tree-based approach [15–18]. Skinny tree (STR) protocol [16] has good performance for member addition. In STR,

$$\text{GK} = g^{x_n g^{x_{n-1} g^{\dots g^{x_3} g^{x_1 x_2}}} \bmod p. \quad (4)$$

While STR uses unbalanced key tree for group key computation, tree-based group Diffie-Hellman (TGDH) leverages balanced tree structure. Given eight group members in TGDH, the group key is computed as follows:

$$\text{GK} = g^{g^{g^{x_1 x_2} g^{x_3 x_4} g^{x_5 x_6} g^{x_7 x_8}}} \bmod p. \quad (5)$$

STR and TGDH require a sponsor node which distributes intermediate computing keys in the tree during membership event changes. As tree-based protocols apparently help to reduce communication cost and operation cost, there have been several variants of TGDH [17, 18]. However, they need to support management for tree balance and require message delivery order due to hierarchical tree structure. In mobile networks, much communication would be required to make sure that the group members can keep the synchronized tree structure.

In summary, DH-based group key protocol is generally known as GKA protocol. Although our protocol is based on DH, we do not classify it as a GKA protocol because of key distribution feature from a controller. Our proposed scheme provides the advantage of dynamics and collaborative contribution in computing group keys with a modified key agreement method.

## 3. Secure Group Key Management for Mobile Networks

*3.1. Membership and Security Requirements.* Group membership events occur with either insertion of a new node or

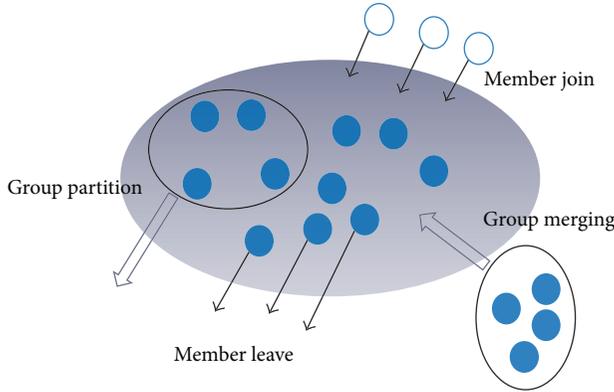


FIGURE 1: Four kinds of membership events; (1) member join (single join or mass join), (2) member leave (a single leave or mass leave), (3) group merging (group join), and (4) group partition (group leave). A small circle represents a node while a big circle represents a group of nodes.

deletion of an existing member. We define the insertion event as *member join* and the deletion event as *member leave*. When there is only one event node specifically, we call each *single join* and *single leave*, and when there are two or more event nodes we call each *mass join* and *mass leave*. Furthermore, we consider a group insertion into a group and a group partition into two distinct groups. We define them as *group merging* and *group partition*, respectively. Figure 1 shows summary of defined membership events.

Group membership change is closely related to security of group communication. Outgoing members should have no access to group communication after it leaves the group, and ingoing nodes should be prevented from accessing previous group communication before it joins the group. We define cryptographic properties in which a secure group, depending on a group key, should meet (1) *group key secrecy* that guarantees an adversary who knows that messages sent to group members cannot discover any group key in polynomial time, (2) *backward secrecy* that guarantees a new member or an adversary who knows that the current group key cannot discover any previous group key in polynomial time, (3) *forward secrecy* that guarantees a former group member or an adversary who knows that previous group keys cannot discover any subsequent group key in polynomial time, (4) *key independence* that guarantees an adversary who knows that a proper subset of group keys cannot discover any other group keys in polynomial time, and (5) (*implicit*) *key authentication* that guarantees that no one apart from a group member recovers the group key.

**3.2. Group Key Establishment.** We present a new group key protocol, collaborative Diffie-Hellman (CODH). CODH has centralized topology and key distribution property from a leader node. But, unlike conventional centralized scheme with TTP, in CODH, a group leader computes and distributes a group key by using public keys of group members. We formalize the group key protocol and prove its security.

CODH has one leader called *master*. The leader is also one of group members. It consumes more energy than normal nodes for communication and operation in managing group keys. There will be a policy for choosing a leader. In mobile networks, signal strength, degree to neighbors, identity, and resources (CPU, memory, battery, and bandwidth) would be criteria for leader election [19–21]. When a group is created, the first master is elected among group members and performs group key initialization. Afterwards, group members select a new master when receiving master notification for leader change. Once a new group master is selected for group management, the previous master forwards information about group members to the new master; that is, a delegation process is run (refer to Sections 3.3 and 3.4). On the other hand, connection failure may occur by network isolation or denial of service attacks. (We assume that group participants are honest and not compromised. However, they can be threatened by network adversaries who can perform all of network-based attacks.) We consider the connection failure as a kind of member leave whether the left node is a member or the master.

Notation section represents notations used to illustrate our group key protocol. The index “*s*” stands for the master node in a group that is distinct from *i* or *j* which indicates a general member node. Therefore,  $M_i$  or  $M_j$  means an identity for general member, while  $M_s$  denotes the master. *Lock-secret* is defined as a secret value of a member. It locks the group key so that  $M_s$  can securely transfer the group key to the members. General members use their *unlock-secret* to extract the group key from  $M_s$ 's broadcast message of a locked group key.

We adopt inverse exponentiation for obtaining the group key. Let  $C_n$  be a group of size  $n$ ; that is,  $C_n = \{M_1, M_2, \dots, M_n\}$  and  $M_s \in C_n$ . To share the initial group key, the group  $C_n$  runs steps in Box 1 for the initial phase.

The initial phase consists of two rounds. In the first round, all members except the group master send their locker  $g^{x_i}$  to the master via unicast and the master produces the locker list,  $XL_C$ , from receiving messages. In the second round, the master  $M_s$  selects a random secret  $k$  and computes and broadcasts the locked group key  $(X_i)^k = (g^{x_i})^k$  using  $XL_C$ . Then, each member can compute the group key GK using their own unlock-secret,  $y_i$ , as follows:

$$\text{GK} \equiv (X_i^k)^{y_i} \pmod{p} \equiv (g^{x_i y_i})^k \pmod{p} \equiv g^k \pmod{p}. \quad (6)$$

The group key is equal to the locker of the group master when  $k$  is the master's secret. Therefore, operations for computing  $X_i^k$  and group messages never include  $X_s$ .

**3.3. Group Rekeying for Member Join and Leave.** The master-secret should be renewed when membership changes, since it is used for the new group key  $\text{GK}'$ . In Box 2 (member join process),  $k'$  means a new master-secret that  $M_s$  selects. Let  $M_{n+1}$  be the first new member and let  $M_{n+m}$  be the last new member, when  $m$  new members join the group  $C_n$  (if a single member joins, the new member is only one node,  $M_{n+1}$ ). A new member  $M_j$  ( $n+1 \leq j \leq n+m$ ) sends its locker  $X_j$  to the

Assume that the group of  $n$  members establish a group key.  
*Step 1.* Each member selects random  $x_i \in \mathbb{Z}_q$  and computes  $X_i = g^{x_i} \bmod p$ .  
 $M_i \rightarrow M_s: X_i$  ( $i \in [1, n], i \neq s$ )  
*Step 2.*  $M_s$  selects random  $k$  in  $\mathbb{Z}_q$  for group key sharing and computes key-locks.  
 $M_s \Rightarrow C_n: \{(X_i)^k \mid i \in [1, n], i \neq s\}$

Box 1: Group key initialization.

Assume that  $m$  members are added to the group  $C_n$ .  
*Step 1.* Each new member  $M_j$  ( $n+1 \leq j \leq n+m$ ) selects random  $x_j \in \mathbb{Z}_q$  and computes  $X_j = g^{x_j} \bmod p$ .  
 $M_j \rightarrow M_s: X_j$  ( $j \in [n+1, n+m]$ )  
*Step 2.*  $M_s$  selects random  $k'$  in  $\mathbb{Z}_q$  for new group key and computes key-locks.  
 $M_s \Rightarrow C_n: \{(X_i)^{k'} \mid i \in [1, n+m], i \neq s\}$

Box 2: Group rekeying for member join.

Assume that a subset  $L_m$  of current group  $C_n$  is composed of  $m$  leaving members in the group and does not include the group master  $M_s$ .  
*Step 1.*  $M_s$  selects random  $k'$  in  $\mathbb{Z}_q$  for new group key and computes key-locks with updated locker list.  
 $M_s \Rightarrow C_n \setminus L_m: \{(X_i)^{k'} \mid i \in [1, n] \wedge M_i \notin L_m, i \neq s\}$

Box 3: Group rekeying for member leave.

master, and then  $M_s$  broadcasts locked new group key  $GK' = g^{k'}$  to all the group members in the same manner as second round of initial phase, as in Box 2. All members, including new members, can extract the new group key  $GK'$  in the same way as (6).

Unlike the join event, member leave process does not require the first round for sending lockers to the master. Let a subset of  $C_n$  for leaving members be  $L_m \subset C_n$  ( $M_s \notin L_m$ ). Group members conduct rekeying operations for the new group key  $GK'$  as in Box 3.

The leaving nodes cannot learn the new group key because the broadcast message from  $M_s$  does not contain any locker  $X_i$  for leaving members. Note that the set  $L_m$  for the leaving node does not include the master. Leaving of the master requires 'delegation' during which the master forwards locker list  $XL_C$  for group  $C_n$  to new group master ( $M_{s'}$ ) as follows:

$$M_s \rightarrow M_{s'}: XL_C = \{X_j \mid M_j \in C_n, j \neq s\}. \quad (7)$$

The delegation can be used for another case where the master wishes to finish its master's role for a reason such as network topology change or resource exhaustion; that is, the master turns to a group member not leaving the group. In this case, the delegation message includes the former master's locker generated with new selected secret  $x_s$  as follows:

$$M_s \rightarrow M_{s'}: XL_C = \{X_j \mid M_j \in C_n, x_s \neq k, x_s \neq k'\}. \quad (8)$$

When group members detect unexpected disconnection from the master, they restart group key initialization with new

master selection. At the worst case, members can suffer from frequent connection failure with the master. In this case, the first protocol should be slightly modified to make all of group members have the locker list and any member be the group master to proceed Box 3. For instance, a general member at the first step of Box 1 broadcasts its locker to the group as follows:

$$M_i \Rightarrow C_n: X_i \text{ } (i \in [1, n], i \neq s). \quad (9)$$

The group members continue secure communication with a fresh group key obtained through group rekeying. We provide formal security proofs in Section 5.

*3.4. Group Rekeying for Group Merging and Partition.* There are two ways to integrate two groups into one group completely: *individual join* and *group join*. The former is that members of a group join another group individually. It is similar to the mass joining process, saving that the joining master should generate his lock-secret,  $x_s$ , and locker,  $g^{x_s}$ . The latter way is that a group is absorbed into the other group via delegation process between both group masters.

Let two groups be merged  $C_n = \{M_1, M_2, \dots, M_n\}$  and  $R_m = \{M_1, M_2, \dots, M_m\}$  ( $n \geq m$ ). The master  $M_s$  of  $C_n$  survives after group merging, while the master  $M_{s'}$  of  $R_m$  becomes a member of the merged group. Smaller group members ( $\in R_m$ ) become a member of  $C_{n+m}$ ; that is,  $C_{n+m} = \{M_1, M_2, \dots, M_n, M_{n+1}, \dots, M_{n+m}\}$  and  $M_s \in C_n$  after group merging. Group merging process runs with delegation (in the first round) as in Box 4. Figure 2 represents an instance for a merging process for a current group  $C_4$  and a merged group

Assume that a group  $R_m$  is merged into a group  $C_n$  where  $n \geq m$ , and the merged group  $C_{n+m} = C_n \cup R_m$ .  $M_{s'}$  is the master of  $R_m$  and  $M_s$  is the master of  $C_n$ .

Step 1.  $M_{s'}$  selects a random number  $x_{s'}$  in  $\mathbb{Z}_q$ , computes  $X_{s'} = g^{x_{s'}} \text{ mod } p$ , and updates the locker list into  $XL_R = \{X_1, X_2, \dots, X_m\} \cup X_{s'}$ .

(delegation)  $M_{s'} \rightarrow M_s: XL_R$

Step 2.  $M_s$  selects random  $k'$  in  $\mathbb{Z}_q$  for new group key and computes key-locks with updated locker list.

$M_s \Rightarrow C_{n+m}: \{(X_i)^{k'} \mid i \in [1, n+m], i \neq s\}$

Box 4: Group merging.

Assume that a current group  $C_n$  is partitioned into two groups,  $P_m$  ( $\subset C_n$ ) and  $C_{n-m}$  ( $= C_n \setminus P_m$ ).

The master of  $R_m$  is  $M_{s'}$  and the master of  $C_n$  is  $M_s$  ( $\notin P_m$ )

Step 1.  $M_s$  generate  $XL_P = \{X_j \mid M_j \in P_m, j \neq s\}$  from  $XL_C$ .

(delegation)  $M_s \rightarrow M_{s'}: XL_P$

Step 2.  $M_s$  and  $M_{s'}$  select random  $k', k''$  in  $\mathbb{Z}_q$  respectively and compute key-locks with their locker list.

$M_s \Rightarrow C_{n-m}: \{(X_i)^{k'} \mid i \in [1, n] \wedge M_i \notin P_m, i \neq s\}$

$M_{s'} \Rightarrow P_m: \{(X_j)^{k''} \mid M_j \in P_m, j \neq s\}$

Box 5: Group partition.

$R_3$ . In Figure 2, the number in a circle indicates members' index (such as by a joined order). Before they are merged, the number of the current group  $C$  is four including the group master (i.e.,  $n = 4$ ,  $C_4$ ) and the number of members of joining group is three (i.e.,  $m = 3$ ,  $R_3$ ). To be merged, the master of  $R_3$  sends the master of  $C_4$  the locker list  $XL_R$  for  $M_1$  and  $M_2$ . Note that the master  $M_s$  of  $R_m$  must forward its locker after changing its own secret because it was used as the former group key. The master of  $C_4$  becomes the master for the merged group. It updates  $XL_C$  and generates key-locks  $XL_C^k$  with a new selected random  $k$ .

As shown in Figure 3, the current group will be divided into two groups. When the number of left members is  $m$ , the current group will have  $(n - m)$  members after the partition process. Group partition requires one more master  $M_{s'}$  for a separated subgroup  $P_m \subset C_n$  ( $M_s \notin P_m$ ). Group partition process can be easily conducted through delegation, from the master  $M_s$  of group  $C_n$  to the fresh master  $M_{s'}$  of subgroup  $P_m$ . The divided groups perform a group key initial phase after the delegation process, as in Box 5.

**3.5. Implicit Key Authentication.** For the secure key authentication, the messages sent from all members should be signed with a signature key. Hash-based signature such as message authentication code (MAC) is fairly efficient in terms of computation cost. However, it is too costly to share one-to-one pairwise keys between all of group members in advance.

We assume that a member holds long-term private and public keys certified by a trusted certificate authority (CA). (Each member can use a different signature algorithm such as RSA-based signature algorithm, digital signature algorithm (DSA), and elliptic curve digital signature algorithm (ECDSA). Note that some of them do not provide message encryption; that is, it is used for message signing and

verifying. We consider that DSA is better for our scheme since its public key includes  $g^x \text{ mod } p$ .) The group members send to the master the signed messages with their own private key; for example, in the first step of Box 1, a member,  $M_i$ , sends to the master  $\{X_i, \text{Sig}_{M_i}(X_i)\}$  which  $M_i$  signs for  $X_i$  with its private key. Note that this process runs one-time at initial phase or it can be precomputed with  $X_i$ .

Members can obtain the group key securely by verifying the messages of the master with signature signed with the master's private key. All of messages from the master come with a master-signed signature for the origin and integrity of a group key. For example, in the second step of Box 1, the master broadcasts  $\{X_1^k, X_2^k, \dots, X_n^k, \text{Sig}_{M_s}(\text{GK})\}$ . The master produces a locked set for the group key using verified members' locker. It implies that outsiders cannot recover the group key from the master's messages.

## 4. Evaluation

We measure performance of the proposed scheme through communication and computation cost spent for all group members to complete group rekeying by membership change. Table 1 shows summary of comparison with other DH-based key management protocols: CKD, GDH, BD, STR, and TGDH. In Table 1,  $n$ ,  $m$ , and  $p$  denote the number of current group members, joining or merged-group members, and leaving or partitioned-group members, respectively. Therefore,  $m = 1$  or  $p = 1$  indicates the single-member event. For TGDH, the height of the key tree is denoted as  $h$ , and, for STR,  $s$  is denoted as the index of the sponsor, which helps other members to calculate group keys. Group merging is a case where a group of  $m$  members is merged into a group of  $n$  members ( $n \geq m$ ), and group partition is a case where a group of  $n$  members is divided into separate subgroups: (1) a group

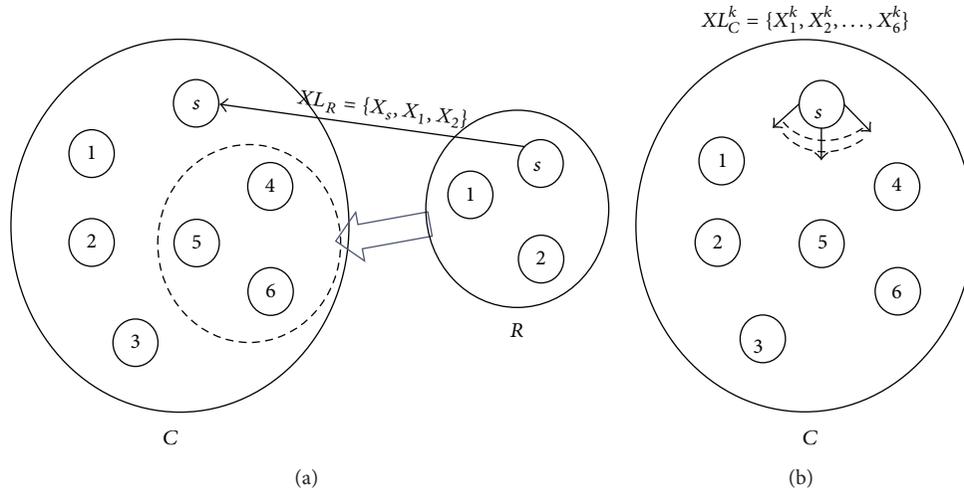


FIGURE 2: Group merging process: (a) when groups  $C_4$  and  $R_3$  are merged,  $R$ 's master sends the locker list of  $R$  to  $C$ 's master and (b) after groups are merged,  $C$ 's master becomes the master for merged group and broadcasts the key-locks for new group key to all of the members.

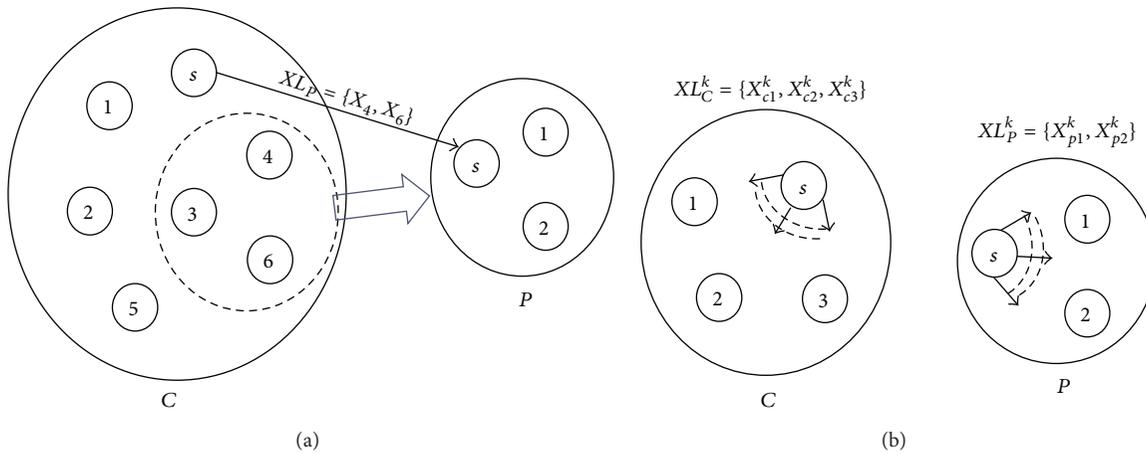


FIGURE 3: Group partition process: (a) when group  $C_7$  is partitioned into two groups (new group  $P_3$ ), the master of the original group sends  $P$ 's master the locker list of the new subgroup and (b) after the group is split, each group master broadcasts the key-locks for each new group key.

of  $p$  members and (2) a group of  $(n - p)$  members, where  $(n - p) \geq p$ . The costs for the group partition event include the costs for updating two subgroup keys. In computation costs, we consider concurrent execution in distributed nodes if it is possible. In CODH, we assume the master is selected by group-join order; the first master is  $M_1$ , and when  $M_1$  leaves the group,  $M_2$  becomes the next master.

CKD distributes the group key in a similar way with our protocol. Its communication and computation costs are also similar to our protocol. However, the worst case of CKD is when the master leaves. It requires large costs for rekeying. On the other hand, in CODH, the rekeying cost for a leaving master is analogous to that for a leaving member due to efficient delegation or sharing of public locker list. GDH is operated through communication chain from the first node

to the last node, and the last node becomes the master of the group. Steiner et al. presented three GDH protocols: GDH.1, 2, and 3. GDH.2 is the most efficient in communication whereas GDH.3 is the most efficient in computation cost among GDH.x. We select GDH.3 for comparison. As shown in Table 1, GDH has weaknesses in group merging and mass joining. BD employs a completely distributed way using broadcast messages. Without sponsors or controllers, all of members broadcast messages for updating the group key. Although it seems to be fairly efficient in computation cost, there are hidden costs for multiplications. In addition, it requires a large communication cost compared to other protocols. STR and TGDH are tree-based key agreement protocols. They use different tree structures for key management. STR, especially, uses the extremely unbalanced tree structure.

TABLE 1: Communication and computation costs.

|      |                  | Rounds  | Messages     | Exponentiations       | Signatures | Verifications |
|------|------------------|---------|--------------|-----------------------|------------|---------------|
| GDH  | Join, merge      | $m + 3$ | $n + 2m + 1$ | $n + 2m + 1$          | $m + 3$    | $n + 2m + 1$  |
|      | Leave            | 1       | 1            | $n - p$               | 1          | 1             |
|      | Partition        | 2       | $p + 1$      | $n - p$               | 2          | $p$           |
|      | Master leave     | 2       | $n - 1$      | $n - 1$               | 2          | $n - 1$       |
| STR  | Join             | 3       | $m + 2$      | $3m$                  | 2          | $m + 2$       |
|      | Leave            | 1       | 1            | $3(n - p) - 2s - 1$   | 1          | 1             |
|      | Merge            | 3       | 3            | $3m - 1$              | 2          | 3             |
|      | Partition        | 2       | 3            | $3(n - p) - 2s - 1$   | 2          | 3             |
| TGDH | Join, merge      | 2       | 3            | $3h - 3$              | 2          | 3             |
|      | Leave, partition | $h$     | $2h$         | $3h$                  | $h$        | $h$           |
| BD   | Join, merge      | 2       | $2n + 2m$    | 3                     | 2          | $2n + 2m - 2$ |
|      | Leave            | 2       | $2n - 2p$    | 3                     | 2          | $2n - 2p - 2$ |
|      | Partition        | 2       | $2n$         | 3                     | 2          | $2n - 2p - 2$ |
| CKD  | Join, merge      | 3       | $m + 2$      | $n + 2m$              | 3          | $m + 2$       |
|      | Leave            | 1       | 1            | $n - p$               | 1          | 1             |
|      | Partition        | 3       | $p + 2$      | $\max(n - p, 2p - 1)$ | 3          | 4             |
|      | Master leave     | 3       | $n$          | $2n - 3$              | 3          | $n$           |
| CODH | Join             | 2       | $m + 1$      | $n + m + 1$           | 2          | $m + 1$       |
|      | Leave            | 1       | 1            | $n - p$               | 1          | 1             |
|      | Merge            | 2       | 2            | $n + m$               | 2          | 2             |
|      | Partition        | 2       | 3            | $n - p$               | 2          | 2             |
|      | Master leave     | 2       | 2            | $n - 1$               | 2          | 2             |

TABLE 2: Communication and computation costs for CODH member and master.

|                |                  | Send | Receive | Exponentiations | Signatures | Verifications |
|----------------|------------------|------|---------|-----------------|------------|---------------|
| General member | Join, merge      | 0    | 1       | 1               | 0          | 1             |
|                | Leave, partition | 0    | 1       | 1               | 0          | 1             |
| Group master   | Join             | 1    | $m$     | $n + m$         | 1          | 1             |
|                | Leave, partition | 1    | 0       | $n - p$         | 1          | 0             |
|                | Merge            | 1    | 1       | $n + m$         | 1          | 1             |

Accordingly, the performance of STR depends on the location of the sponsors. In TGDH, the costs depend on the height of the resulting key tree and locations of joining or leaving members in the tree. We provide the worst case cost for TGDH.

Most of the cost in CODH comes from the master node. A general node consumes only one communication, modular exponentiation, signature, and verification in all of group rekeying process. We summarize the costs for a general member and the group master in Table 2. Although the exponentiation cost looks heavy in the master, its cost is insignificant. We conducted an experiment to measure computation delays for modular exponentiations. Table 3 shows the average delay of 10 experimental results for each. The first device has less CPU power than the second device. When modular prime  $p$  is 1024 bits long and  $n \leq 50$ , the computation delay is less than 1 s. The average delay of one

exponentiation is less than 8 ms in the second device. Moreover, reducing communication cost is important for mobile devices because data communication consumes more energy than any other process. Therefore, our group key protocols can be efficiently applied in dynamic mobile networks.

## 5. Security

Let  $p$  be a large prime number of the form  $2q + 1$  for a prime  $q$  in  $\mathbb{Z}_p$ . Let  $G$  be a cyclic group of prime order  $q$  and let  $g$  be a generator of  $G$ ; that is,  $G = \langle g \rangle$ . The decisional Diffie-Hellman problem (DDH) is as follows: given  $(g, g^x, g^y, g^z)$ , where  $x, y, z \in \mathbb{Z}_q$ , decide whether  $z = xy$  or a randomly chosen number. In particular, the security of our protocol is based on the divisible decisional Diffie-Hellman problem

TABLE 3: Computation delays on mobile devices (ms).

|                                         | $p = 1024\text{-bit}$ |          |          | $p = 2048\text{-bit}$ |          |          |
|-----------------------------------------|-----------------------|----------|----------|-----------------------|----------|----------|
|                                         | $n = 1$               | $n = 25$ | $n = 50$ | $n = 1$               | $n = 25$ | $n = 50$ |
| Exponentiation (1 GHz CPU, 512 MB RAM)  | 37                    | 452.6    | 907.8    | 168.9                 | 3385.4   | 6762.5   |
| Exponentiation (2.26 GHz CPU, 2 GB RAM) | 12.9                  | 192.8    | 385.3    | 75.4                  | 1458.8   | 2917.4   |

(DDDH), which is stronger assumption than the divisible computational Diffie-Hellman problem (DCDH).

*Definition 1.* The DCDH problem is as follows: given  $(g, g^x, g^y)$ , where  $x, y \in \mathbb{Z}_q$ , compute  $g^{y/x}$ .

*Definition 2.* The DDDH problem is as follows: given  $(g, g^x, g^y, g^z)$ , where  $x, y, z \in \mathbb{Z}_q$ , decide whether  $z = y/x$  or a randomly chosen number.

The DDDH problem is weaker than DCDH, since if an adversary could solve the DCDH problem, he could solve the DDDH problem by computing  $g^x$  to decide  $g^z = g^{y/x}$ ; thus the DDDH assumption is stronger than the DCDH assumption. Similarly, the DDH problem is weaker than the computational Diffie-Hellman problem (CDH), which is weaker than discrete logarithm problem (DL) [22]. We want to prove the security of our protocol under the DDH and DDDH assumptions.

**Theorem 3.** *The DDDH problem is equivalent to the DDH problem.*

*Proof.* Given the DDDH input  $(g, g^x, g^y, g^z)$ , where  $z = y/x$ , one submits  $(g, g^x, g^z, g^y)$  to DDH to decide whether  $y = xz$  or a randomly chosen number. Similarly, given the DDH input  $(g, g^x, g^y, g^z)$ , where  $z = xy$ , one submits  $(g, g^x, g^z, g^y)$  to DDDH to decide if  $y = z/x$  or a randomly chosen number.  $\square$

Therefore, we know that if there is no polynomial time algorithm to solve the DDH problem, it is hard to solve the DDDH problem.

**Theorem 4.** *If the DDH problem is hard, it is hard to find a polynomial time algorithm to recover the group key from the proposed protocol; in other words, it provides group key secrecy against passive adversaries under the DDH assumption.*

*Proof.* Let  $view(n, k)$  be public information for a group of  $n$  members to establish a group key  $g^k$ ; thus it is a view of passive attackers,

$$view(n, k) := (g^{x_1}, g^{x_1 k}, g^{x_2}, g^{x_2 k}, \dots, g^{x_n}, g^{x_n k}). \quad (10)$$

Suppose we had an algorithm  $F$  that with significant probability succeeds to distinguish between  $(view(n, k), g^y)$ , where  $y$  is a random number  $y \in \mathbb{Z}_q$ , and  $(view(n, k), g^k)$  where  $g^k$  is the group key; that is,  $F(view(n, k), g^y) = F(g^{x_1}, g^{x_1 k}, g^{x_2}, g^{x_2 k}, \dots, g^{x_n}, g^{x_n k}, g^y) = 1$ , where  $y = k$ ,

otherwise, returns 0. Then we can query to  $F$  with input  $view(n-1, k) = (g^{x_1}, g^{x_1 k}, g^{x_2}, g^{x_2 k}, \dots, g^{x_{n-1}}, g^{x_{n-1} k})$  for  $n-1$  members' information and additional input  $((g^{x_i})^r, (g^{x_i k})^r)$  for a random number  $r \in_{\mathbb{R}} \mathbb{Z}_q$ , where  $0 < i < n$ , that is,  $F(view(n-1, k), g^{x_i r}, g^{x_i k r}, g^y)$ . It follows that  $(view(1, k), g^{x r_1}, g^{x k r_1}, g^{x r_2}, g^{x k r_2}, \dots, g^{x r_{n-1}}, g^{x k r_{n-1}}, g^y) = F(g^x, g^{x k}, g^{x r_1}, g^{x k r_1}, g^{x r_2}, g^{x k r_2}, \dots, g^{x r_{n-1}}, g^{x k r_{n-1}}, g^y)$ , where  $r_i \in_{\mathbb{R}} \mathbb{Z}_q$  for  $0 < i < n$ . Then  $F$  can solve the DDDH problem since it can decide whether  $y = xk/x$  or a random number, given  $(view(1, k), g^y) = (g^x, g^{x k}, g^y)$ . It means that  $F$  can also solve the DDH problem by Theorem 3.  $\square$

**Theorem 5.** *The proposed scheme provides backward secrecy, forward secrecy, and key independence provided the DDH problem is intractable.*

*Proof.* Whenever membership is changed or the group key is updated, the group controller alters its own secret  $k$  to  $k'$ , where  $k'$  is an independently random number to  $k \in \mathbb{Z}_q$ ; it implies that it is impossible to find an algorithm  $F$  such that  $F(g^k) \rightarrow g^{k'}$  without knowledge of  $k$  and  $k'$ . We assume that the secret values are uniformly distributed by a pseudorandom generator. Therefore, when the group key has been changed, an adversary must use new public information,  $view(n, k') = (g^{x_1}, g^{x_1 k'}, g^{x_2}, g^{x_2 k'}, \dots, g^{x_n}, g^{x_n k'})$ , to recover the group key updated into  $g^{k'}$  and it depends on a solution to solve the DDH problem by Theorem 4. It follows that past members, future members, or adversaries who know a subset of previous group keys cannot learn the current group key, since the broadcast message from the master does not contain their locker  $X_i$  in  $view()$ .  $\square$

**Theorem 6.** *The proposed scheme provides implicit key authentication under the security of certified public key.*

*Proof.* A locker which the master obtains from group members is what a group member signs with its public key certified by a CA. Concretely, a locker  $X_i$  is hashed by a one-way function such as SHA-2, and hash  $(X_i)$  is signed with  $M_i$ 's private key using a digital signature algorithm such as RSA, DSA, and ECDSA. Then, the locker is verified with the public key bound to  $M_i$  and certified by CA. If there is a locker of nonmember in the locker list of a group, it must be along with a forged signature. It means that the problem occurs in a hash collision attack or a rogue CA certificate [23]. Once all verified lockers are transferred to the master, any other nodes which are not a group member cannot recover the group key under the DDH assumption (Theorems 4 and 5).  $\square$

## 6. Conclusion

In this paper, we propose a secure group key management protocol based on DH key agreement. The proposed key management requires only one data communication and one modular exponentiation at each member for any membership event. It shows prominent efficiency in renewing the group keys against dynamic group membership change, member join/leave and group merging/partition. We proved group key secrecy, backward/forward secrecy, key independence, and key authentication. No outsiders can learn the group key under the DDH assumption. We conclude that CODH can be adapted efficiently for multicast security in mobile networks.

## Notations

|                            |                                                                                                   |
|----------------------------|---------------------------------------------------------------------------------------------------|
| $n$ :                      | Number of protocol participants                                                                   |
| $M_i$ :                    | $i$ th group member, $i \in [1, n]$                                                               |
| $M_s$ :                    | Master node (controller), $s \in [1, n]$                                                          |
| $p$ :                      | Prime of the form $2q + 1$ for a prime $q$                                                        |
| $g$ :                      | Generator in $\mathbb{Z}_p^*$                                                                     |
| $x_i$ :                    | Lock-secret; random number picked by $M_i$ such that $1 < x_i < p - 1$ and $\gcd(x_i, p - 1) = 1$ |
| $y_i$ :                    | Unlock-secret for $M_i$ such that $x_i * y_i \equiv 1 \pmod{p - 1}$                               |
| $k$ :                      | Master-secret randomly selected in $\mathbb{Z}_q^*$ , by $M_s$                                    |
| $X_i$ :                    | Locker; $g^{x_i} \pmod{p}$                                                                        |
| $C_n$ :                    | Current group of $n$ members; $\#(C) = n$                                                         |
| $XL_C$ :                   | Locker list of group $C$ ; $XL_C = \{X_1, X_2, \dots, X_n\} \setminus X_s$                        |
| $XL_C^k$ :                 | Key-locks for group $C$ ; $XL_C^k = \{X_1^k, X_2^k, \dots, X_n^k\} \setminus X_s^k$               |
| $M_i \rightarrow M_j$ : m: | Unicast message (m) from $M_i$ to $M_j$                                                           |
| $M_i \Rightarrow C_n$ : m: | Broadcast message (m) from $M_i$ to $n$ members of $C$ .                                          |

## Conflict of Interests

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

## Acknowledgments

The authors appreciate anonymous reviewers for their helpful comments. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0011289).

## References

- [1] R. Canetti, T. Malkin, and K. Nissim, "Efficient communication-storage tradeoffs for multicast encryption," in *Proceedings of Advances in Cryptology (Eurocrypt '99)*, vol. 1592 of *Lecture Notes in Computer Science*, pp. 459–474, Prague, Czech Republic, May 1999.
- [2] S. Setia, S. Koussih, S. Jajodia, and E. Harder, "Kronos: a scalable group re-keying approach for secure multicast," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 215–228, Berkeley, Calif, USA, May 2000.
- [3] M. K. Reiter, "A secure group membership protocol," *IEEE Transactions on Software Engineering*, vol. 22, no. 1, pp. 31–42, 1996.
- [4] D. Wallner, E. Harder, and R. Agee, "Key management for multicast: issues and architectures," RFC 2627 Informational, 1999.
- [5] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM Transactions on Networking*, vol. 8, no. 1, pp. 16–30, 2000.
- [6] S. Mitra, "Iolus: a framework for scalable secure multicasting," in *Proceedings of the ACM (SIGCOMM '97)*, pp. 277–288, Cannes, France, September 1997.
- [7] A. T. Sherman and D. A. McGrew, "Key establishment in large dynamic groups using one-way function trees," *IEEE Transactions on Software Engineering*, vol. 29, no. 5, pp. 444–458, 2003.
- [8] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 62–72, Washington, DC, USA, October 2003.
- [9] S. Zhu, S. Xu, S. Setia, and S. Jajodia, "Establishing pairwise keys for secure communication in ad hoc networks: a probabilistic approach," in *Proceedings of the 11th IEEE International Conference on Network Protocols*, pp. 326–335, Atlanta, Ga, USA, 2003.
- [10] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.
- [11] M. Burmester and Y. Desmedt, "A secure and scalable group key exchange system," *Information Processing Letters*, vol. 94, no. 3, pp. 137–143, 2005.
- [12] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman key distribution extended to group communication," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, pp. 31–37, New Delhi, India, March 1996.
- [13] M. Steiner, G. Tsudik, and M. Waidner, "CLIQUES: a new approach to group key agreement," in *Proceedings of the 18th International Conference on Distributed Computing Systems*, pp. 380–387, May 1998.
- [14] Y. Amir, Y. Kim, C. Nita-Rotaru, and G. Tsudik, "On the performance of group key agreement protocols," *ACM Transactions on Information and System Security*, vol. 7, no. 3, pp. 457–488, 2004.
- [15] Y. Kim, A. Perrig, and G. Tsudik, "Tree-based group key agreement," *ACM Transactions on Information and System Security*, vol. 7, no. 1, pp. 60–96, 2004.
- [16] Y. Kim, A. Perrig, and G. Tsudik, "Group key agreement efficient in communication," *IEEE Transactions on Computers*, vol. 53, no. 7, pp. 905–921, 2004.
- [17] B. Wu, J. Wu, and D. Yuhong, "An efficient group key management scheme for mobile ad hoc networks," *International Journal of Security and Networks*, vol. 4, no. 1-2, pp. 125–134, 2009.
- [18] P. P. C. Lee, J. C. S. Lui, and D. K. Y. Yau, "Distributed collaborative key agreement and authentication protocols for dynamic peer groups," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 263–276, 2006.

- [19] J. Liu, D. Sacchetti, F. Sailhan, and V. Issarny, "Group management for mobile Ad Hoc networks: design, implementation and experiment," in *Proceedings of the 6th International Conference on Mobile Data Management (MDM '05)*, pp. 192–199, Ayia Napa, Cyprus, May 2005.
- [20] B. Singh and D. K. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, no. 13, pp. 1–18, 2012.
- [21] C.-W. Chen, Y.-R. Tsai, and S.-J. Wang, "Cost-saving key agreement via secret sharing in two-party communication systems," *Journal of Convergence*, vol. 3, no. 4, pp. 29–36, 2012.
- [22] S. Mohanty and B. Majhi, "A strong designated verifiable dl based signcryption scheme," *Journal of Information Processing Systems*, vol. 8, no. 4, pp. 567–574, 2012.
- [23] M. Stevens, A. K. Lenstra, and B. de Weger, "Chosen-prefix collisions for MD5 and applications," *International Journal of Applied Cryptography*, vol. 2, no. 4, pp. 322–359, 2012.

## Research Article

# Modeling and Implementing Two-Stage AdaBoost for Real-Time Vehicle License Plate Detection

**Moon Kyou Song and Md. Mostafa Kamal Sarker**

*Department of Electronics Convergence Engineering, Wonkwang University, 344-2 Shinyong Dong, Iksan, Jeonbuk 570-749, Republic of Korea*

Correspondence should be addressed to Md. Mostafa Kamal Sarker; [mksarker@wku.ac.kr](mailto:mksarker@wku.ac.kr)

Received 6 March 2014; Accepted 31 July 2014; Published 18 August 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 M. K. Song and Md. M. K. Sarker. 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.

License plate (LP) detection is the most imperative part of the automatic LP recognition system. In previous years, different methods, techniques, and algorithms have been developed for LP detection (LPD) systems. This paper proposes to automatic detection of car LPs via image processing techniques based on classifier or machine learning algorithms. In this paper, we propose a real-time and robust method for LPD systems using the two-stage adaptive boosting (AdaBoost) algorithm combined with different image preprocessing techniques. Haar-like features are used to compute and select features from LP images. The AdaBoost algorithm is used to classify parts of an image within a search window by a trained strong classifier as either LP or non-LP. Adaptive thresholding is used for the image preprocessing method applied to those images that are of insufficient quality for LPD. This method is of a faster speed and higher accuracy than most of the existing methods used in LPD. Experimental results demonstrate that the average LPD rate is 98.38% and the computational time is approximately 49 ms.

## 1. Introduction

Within the last few decades, LP recognition (LPR) has become an extremely popular and active research topic in the image processing domain. With the constant increase of traffic on the roads, there is a need for intelligent traffic management systems that can detect and track a vehicle as well as identify it. Most of the previous LP detection (LPD) algorithms are restricted in certain working conditions, such as fixed backgrounds [1], known color [2], or fixed size of the LPs [3]. Therefore, detecting LPs under various complex environments remains a challenging problem.

In this paper, we evaluate how well object detection methods used in text extraction [4] and face detection [5] apply to the problem of LP detection. We present a novel method for locating the LP rapidly using the two-stage cascade AdaBoost combined with different image preprocessing procedures. The cascade AdaBoost has two phases in two stages, offline training and online detection.

In the first stage of the cascade AdaBoost, the size of positive samples is extremely important for offline training;

consequently, all positive images should be the same size. Using boundary padding or boundary pixel extension [6], the training positive sample images are created of the same size. Image preprocessing is organized with a Sobel vertical operator applied to the edge of the image; the image edge is then smoothed using the Gaussian filter. Once preprocessing is finished, the AdaBoost training phase starts. After the training process is complete, the detection phase becomes ready to detect the LP. During the online detection phase, the original images are resized for faster detection, and the same image preprocessing methods as in the offline training phase are applied. If the LP is detected with the trained cascade, the detected LP is verified through the connected component analysis (CCA). If this stage detects the LP correctly, the LP is saved; otherwise, the LP image is sent to the next stage.

In the second stage, all procedures are similar to the first stage of the cascade AdaBoost, except for the image preprocessing techniques. In this second stage, we find that most of the poor quality images that are rejected from the first stage have variant illumination, blurring, ambient lighting conditions, and so forth. Therefore, we use adaptive

| Type           |            | Before<br>(2006/11/1) | After<br>(2006/11/1) |  |
|----------------|------------|-----------------------|----------------------|--|
| Regular        | Personal   |                       |                      |  |
|                | Commercial |                       |                      |  |
| Large vehicles | Personal   |                       |                      |  |
|                | Commercial |                       |                      |  |
| Rental cars    |            |                       |                      |  |

FIGURE 1: Different types and sizes of Korean LPs.

thresholding to obtain maximum edge information from those images. The detection results obtained in this stage are remarkable for poor quality images. After finishing the two-stage cascade AdaBoost, we find that the average LPD rate is 98.38% with a computational time of 49 ms.

This paper is organized as follows. Background and challenges are illustrated in Section 2, and our proposed LPD method is described in Section 3. The experimental results in Section 4 show that the proposed method is able to ensure fast LPD as well as achieve sufficient accuracy. Finally, the conclusion is summarized in Section 5.

## 2. Background and Challenges

To properly work with LPR systems, we must manage a large variety of LPs, especially in South Korea. Each province in Korea has its own LP color, pattern, and formats of numbers and other characters. Different colors represent different types of vehicles. Moreover, there are three different sizes of LPs available in Korea, such as large (520 mm × 110 mm), medium (440 mm × 200 mm), and small (335 mm × 170 mm or 155 mm). Figure 1 shows the different types and sizes of LPs available before and after November 01, 2006, in Korea.

## 3. Proposed System

Our proposed LPD system consists of two parts; the first part uses cascade AdaBoost and the second part uses adaptive thresholding. (See Figure 2 for the system architecture of our proposed system.)

**3.1. Using Cascade AdaBoost.** Using cascade AdaBoost for LPD systems consists of two phases, offline training and online detection, as shown in Figure 3.

**3.1.1. Offline Training Phase.** At the core of the offline training phase are the training and combination of strong classifiers. First, a series of weak classifiers (critical features) with their weights are extracted after being trained by a large number of positive and negative examples. Then, strong classifiers are selected from the weak classifiers according to their weights. Figure 4 shows how the algorithm is structured. The strong

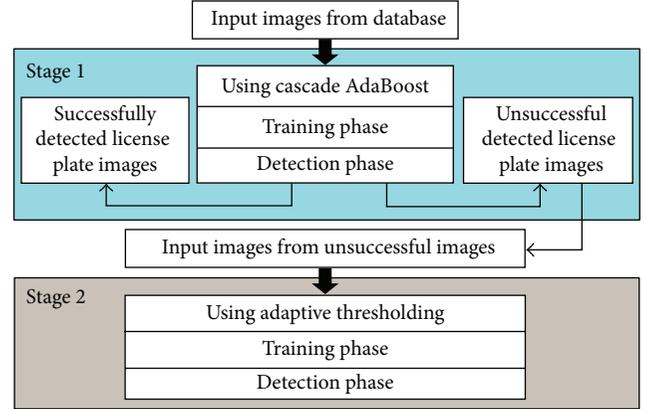


FIGURE 2: System architecture of the proposed LPD system using cascade AdaBoost.

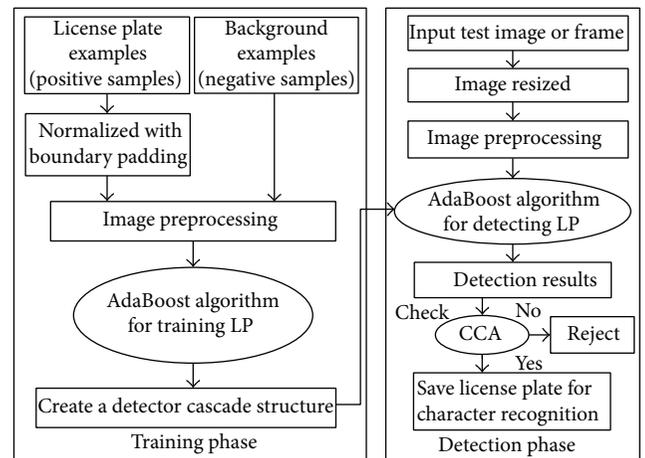


FIGURE 3: Framework of the proposed LPD system using cascade AdaBoost.

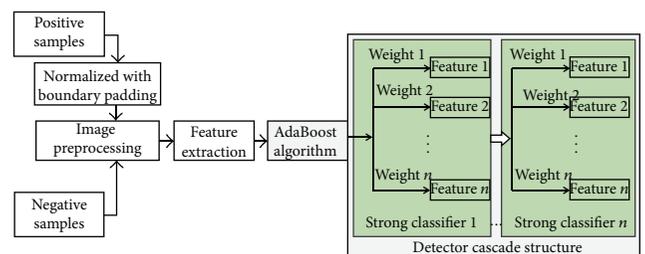


FIGURE 4: Structure for the offline training phase.

classifiers are then constructed in a detector cascade structure for the online recognizing module.

(1) *The Training Databases.* For offline training, positive sample images and negative sample images are required. The positive sample images are LP images only; the negative sample images are background images without an LP image.

(a) *The Positive Samples.* The Korean LP is made of three different sizes, large (520 mm × 110 mm), medium



FIGURE 5: Positive sample images of Korean LPs.

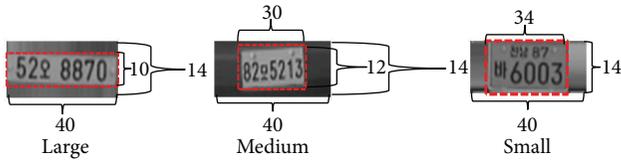


FIGURE 6: Normalized LP images of positive samples with boundary padding.

(440 mm × 200 mm), and small (335 mm × 170 mm or 155 mm). A total of 15,000 images are used as the positive sample images (6,000 large; 3,000 medium; and 6,000 small) for our training experiment. Figure 5 shows some positive sample images of Korean LPs.

The AdaBoost training algorithm requires that the positive sample images be of the same size. For this reason, we must normalize all three types of Korean LP images into one equal size. Boundary padding or boundary pixel extension [6] causes all the positive sample images to be of the same size. For our training case, an image size resolution of 40 × 14 is applied because all the LP regions of our database images are under this resolution. Figure 6 shows the normalized LP images with boundary padding or boundary pixel extension.

(b) *The Negative Samples.* The negative sample images should appear without the LP; for example, they can be images of a part of the car, the road, trees, and so forth. In our training procedure, a total of 25,000 images are used for the negative samples. Figure 7 shows some negative sample images.

(2) *Image Preprocessing.* In this section, we describe the image converting, filtering, and edge detection methods that are applied to preprocess the training images.

(a) *Image Converting.* Given that RGB images have more depth, they are difficult to process; therefore, we convert the original images into gray scale images. In addition, image enhancement and edge detection are performed on the gray scale images in order to adjust the structural property of the images in preparation for LP region detection. The input

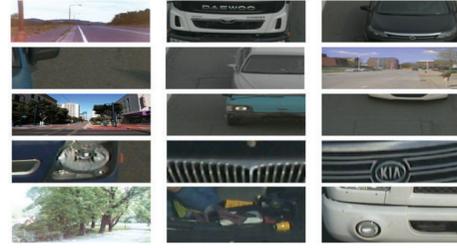


FIGURE 7: Negative sample images.

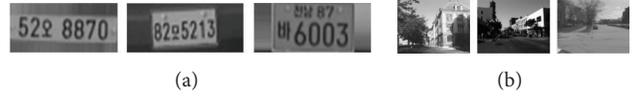


FIGURE 8: Examples of filtered images after using Gaussian filter: (a) positive samples and (b) negative samples.

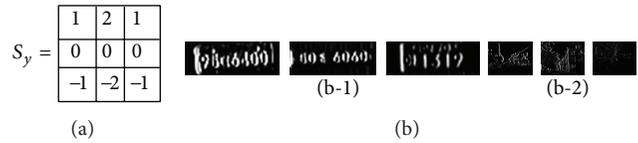


FIGURE 9: (a) Convolution mask of Sobel vertical edge operator, (b) examples of edge images after using Sobel vertical edge operator (b-1) positive samples and (b-2) negative samples.

images are converted from 24-bit color images to 8-bit gray scale images using

$$\text{Gray value} = 0.3 * \text{Red} + 0.59 * \text{Green} + 0.11 * \text{Blue}. \quad (1)$$

(b) *Image Filtering.* Gaussian filter is applied for image filtering. Gaussian filter is the weighted averaging of neighboring pixels; the weights are chosen according to the shape of a Gaussian function, which is defined as

$$g [i, j] = e^{-\frac{(i^2 + j^2)}{2\sigma^2}}, \quad (2)$$

where  $i$  is the distance from the origin in the horizontal axis,  $j$  is the distance from the origin in the vertical axis, and  $\sigma$  is the standard deviation of the Gaussian distribution. Figure 8 shows the filtered images using Gaussian filter.

(c) *Edge Detection (Edge Image).* For edge detection or edge image, the Sobel vertical edge operator [7] is applied. Figure 9 defines the convolution mask of the Sobel vertical edge operator and the edge image after using the Sobel vertical edge operator.

(3) *Feature Extraction.* LP location procedures classify images based on the value of simple features by using the intensity values of a pixel. These features are using the change in contrast values between adjacent rectangular groups of pixels.

The contrast variances between the pixel groups are used to determine relative light and dark areas. Two or three adjacent groups with a relative contrast variance form a Haar-like feature. These features are used for the LPD shown in Figure 10.

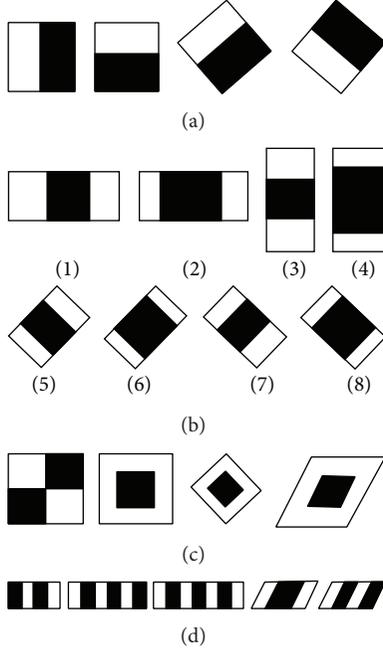


FIGURE 10: Haar-like prototypes used in our algorithm: (a) edge features, (b) line features, (c) center-surrounding features, and (d) plate character features.

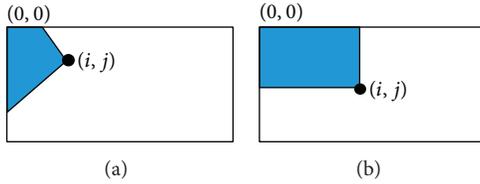


FIGURE 11: (a) Summed area of integral image and (b) summed area of rotated integral image.

By using a transitional depiction of an image, the simple rectangular features of an image are calculated. This is called the integral image [8]. The integral image is an array that contains the sums of the pixel intensity values located directly to the left of a pixel and directly above the pixel at location  $(i, j)$ , inclusively. Therefore, if  $O[i, j]$  is the original image and  $OI[i, j]$  is the integral image, the integral image is calculated as shown in (3) and demonstrated in Figure 11:

$$OI[i, j] = \sum_{i' \leq i, j' \leq j} O(i', j'). \quad (3)$$

The features are rotated 45 degrees, similar to the line feature shown in Figure 10(b)(5), as presented by Lienhart and Maydt [9]. Such features require another transitional depiction called the rotated integral image or rotated sum auxiliary image. The rotated integral image is computed by finding the sum of the pixel intensity values that are located at a 45-degree angle to the left and above of the  $i$  value and below the  $j$  value. Therefore, if  $O[i, j]$  is the original image

and  $OR[i, j]$  is the rotated integral image, the integral image is calculated as shown in (4) and illustrated in Figure 11:

$$OR[i, j] = \sum_{i' \leq i, i' \leq i - |j - j'|} O(i', j'). \quad (4)$$

(4) *AdaBoost Algorithm for Training LP.* The cascade boosted classifier that is created in the Haar-like features training process for several LP samples locates the LP extremely fast and correctly. For each feature, the weak learner determines the optimal threshold classification function, such that the minimum number of examples is misclassified. Thus, a weak classifier  $h(x, f, p, \theta)$  consists of a feature ( $f$ ), a threshold ( $\theta$ ), and a polarity ( $p$ ) that indicates the direction of the inequality sign:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } pf(x) < p\theta \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

The boosting process and the AdaBoost algorithm for classifier learning are as follows.

- (1) Give sample images  $(x_1, y_1) \cdots (x_n, y_n)$ , where  $y_i = 0$  and 1 for negative and positive samples, respectively.
- (2) Initialize weights  $\omega_1$ ,  $i = 1/2 m$  and  $1/2 l$  for  $y_i = 0$  and 1, respectively, where  $m$  and  $l$  are the number of negatives and positives, respectively.
- (3) For  $t = 1 \cdots T$ ,

- (a) normalize the weights

$$\omega_{t,i} = \frac{\omega_{t,j}}{\sum_{j=1}^n \omega_{t,j}} \quad (6)$$

such that  $\omega_t$  is a probability distribution.

- (b) Select the best weak classifier with respect to the weighted error

$$\epsilon_t = \min_{f,p,\theta} \sum \omega_i |h(x_i, f, p, \theta) - y_i|. \quad (7)$$

- (c) Define  $h_t(x) = h(x, f_t, p_t, \theta_t)$ , where  $f_t, p_t$ , and  $\theta_t$  are the minimizers of  $\epsilon_t$ .
- (d) Update the weights

$$\omega_{t+1,i} = \omega_{t,i} \beta_t^{1-e_i}, \quad (8)$$

where  $e_i = 0$  if example  $x_i$  is classified properly; otherwise,  $e_i = 1$  and  $\beta_t = e_t / (1 - e_t)$ .

- (4) The final strong classifier is

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

where  $\alpha_t = \log 1/\beta_t$ .

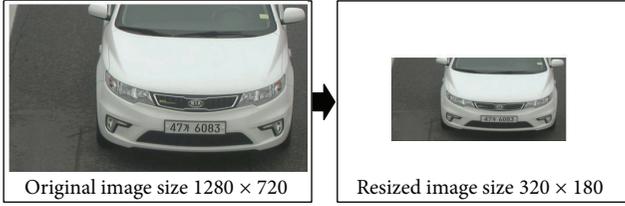


FIGURE 12: Resized image from the original image.

The methods used involve training a strong classifier using the AdaBoost algorithm. Over numerous sequences, AdaBoost chooses the best performing weak classifier from a group of weak classifiers acting on a single feature; once trained, AdaBoost combines the respective votes of the classifiers in a weighted manner, thus forming a strong classifier. This strong classifier is then applied to the subregions of the image that is being scanned for possible LP locations. The weak learning algorithm is designed to select the single rectangle feature that best separates the positive and negative samples. An optimization introduced by Viola and Jones [8] involves a cascade of strong classifiers, each with precisely designed false-positives and false-negatives rates, that greatly speeds up the scanning process because not all classifiers must be evaluated to exclude most non-LP subregions. A background threshold of 80, a number of training stages of 14, and a total number of features of 61,789 are used in our AdaBoost training phase. After finishing the training procedure of the AdaBoost algorithm, a detector cascade structure is created as an XML file. This XML file contains the strong classifier with features.

**3.1.2. Online Detection Phase.** For the online LPD, the number of test images is 1,800 with a resolution of  $1280 \times 720$ . (See Section 4.1 for an explanation of the databases.) The details of the online LPD procedure are described next.

(1) *Resizing the Input Images.* The size of the images in our databases is extremely large. A large image resolution requires more computational time; therefore, we resized the original test images ( $1280 \times 720$ ) into a resolution of  $320 \times 180$  to accelerate the detection procedure. Figure 12 shows an image resized from the original.

(2) *Image Preprocessing.* The same image preprocessing techniques explained in Section 3.1.1(2) are utilized in the offline training phase: first, convert the images from RGB to gray scale; then, filter the images with Gaussian filter; finally, apply edge detection (edge image) with the Sobel vertical edge operator. Figure 13 shows the results of image preprocessing.

(3) *AdaBoost Algorithm for Detecting LPs.* The strong classifiers combine with each other to form a classifier cascade. The strong classifier from the first layer allows a vast majority of the image regions to be recognized and passed to the next layer; at the same time, the classifier rejects as many negative samples as possible. Thus, the classifier cascade has stronger classification abilities, and the final result is more likely to be an LP. Figure 14 shows the cascade structure of the LPD.

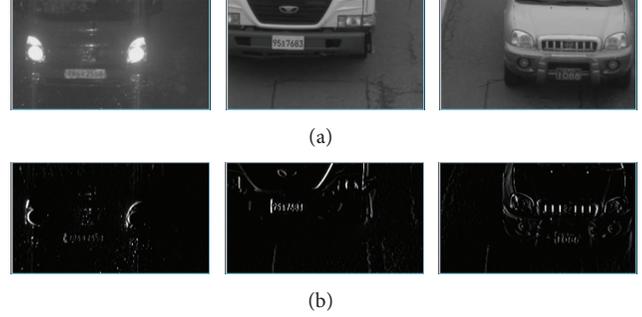


FIGURE 13: Results of image preprocessing: (a) examples of filtered image after using Gaussian filter and (b) examples of edge image after using Sobel vertical operator.

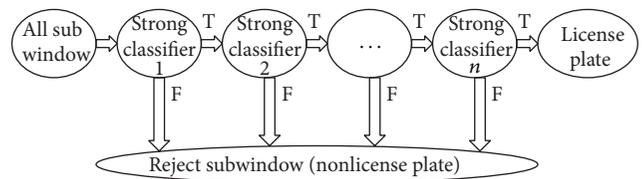


FIGURE 14: Cascade AdaBoost structure for LPD.

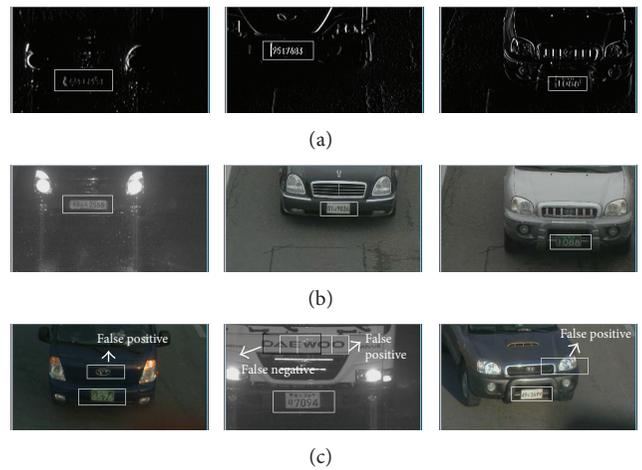


FIGURE 15: Results of LPD using cascade AdaBoost: (a) examples of LPD with edge images, (b) examples of LPD with resized images, and (c) examples of LPD with false positives/negatives.

During the combination process, the strong classifier that consists of more important features and an easier structure is placed at the top of the entire classifier cascade in order for the system to exclude as many negative samples as possible, thus accelerating the detection of LPs.

(4) *LPD Results.* Figure 15 shows the results of the LPD using our proposed cascade AdaBoost algorithm.

(5) *Verify Detected LP Images with CCA.* There are many false positives/negatives areas detected as LP regions using the cascade AdaBoost algorithm. To ignore such false positives/negatives, we use CCA. Figure 16 shows the procedure

TABLE 1: LPD using cascade AdaBoost.

| Number of test images | False positives/negatives |                    | Test accuracy | Time  |
|-----------------------|---------------------------|--------------------|---------------|-------|
|                       | Before applying CCA       | After applying CCA |               |       |
| 1800                  | 17%                       | 0%                 | 87.94%        | 45 ms |

- (1) If number of Blob  $\geq 6$  and  $\leq 10$
  - (2)     area = LP
  - (3) Else
  - (4)     area = Non-LP
  - (5) End

PROCEDURE 1: Blob (number, area).

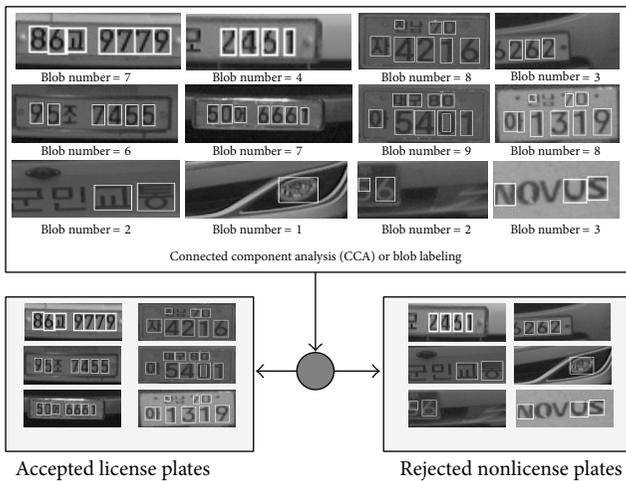


FIGURE 16: Verifying images with CCA and saving LP images.

for verifying detected LP images with CCA. The verification LP and non-LP procedure is as in Procedure 1.

**3.2. Using Adaptive Thresholding.** Using adaptive thresholding for LPD systems is similar as using the cascade AdaBoost algorithm, (see Section 3.1), with the exception of the image preprocessing method. The majority of the images rejected from the first stage have variant illumination, blurring, ambient lighting conditions, and so forth. Therefore, for the image preprocessing technique in this stage, we use adaptive thresholding to compare each pixel of an image to an average of the surrounding pixels. The procedure for adaptive thresholding is as in Procedure 2.

Figure 17 shows the LPD results after applying adaptive thresholding.

## 4. Experimental Results

To test the performance of our proposed method, we use our own database with 1,800 images. The details of the experimental results are presented next.



FIGURE 17: LPD results using adaptive thresholding.

**4.1. Databases.** For our test experiments, we used one database to calculate the detection rate and the computational time. The numbers of total images are 1,800. All the images in our database are rotated and illuminated; furthermore, the images were captured using a CMOS camera under different weather conditions.

**4.2. Experimental LPD Results Using Cascade AdaBoost.** To test the LPD using cascade AdaBoost method proposed in this paper, we applied the method to a database of 1,800 images that were captured at different times and weather conditions. The experiment is based on the conditions of a system with CPU 3.10-GHz Intel Core i3-2100 and 4.00 GB of RAM and implemented using Microsoft Visual Studio 2010 with OpenCV library. Table 1 lists the LPD rate, the percentage of false positives/negatives, and the computational time with the database.

From Table 1, we can see that the total number of detected images is 1,583 and the number of detected false positives/negatives is 306. After applying CCA, no false positives/negatives remained.

**4.3. Experimental Results for LPD Using Adaptive Thresholding.** The remaining 217 images that were not detected correctly (from step one of the phase that uses the cascade AdaBoost algorithm) are used as input images in this phase, adaptive thresholding is applied to them, and then the images are used for training and testing with the same procedures employed for the cascade AdaBoost phase.

From Table 2, we can see that the total number of detected images is 188, and the number of detected false positives/negatives is 35. After applying CCA, no false positives/negatives remain. The images that were not detected properly are 29.

**4.4. Summary of Experimental Results.** The total number of test images is 1,800. The total number of detected images is 1,771. The number of images that could not be detected

```

(1) for $p = 0$ to w do
(2) sum $\leftarrow 0$
(3) for $q = 0$ to h do
(4) sum \leftarrow sum + in[p, q]
(5) if $p = 0$ then
(6) intImg[p, q] \leftarrow sum
(7) else
(8) intImg[p, q] \leftarrow intImg[$p - 1, q$] + sum
(9) end if
(10) end for
(11) end for
(12) for $p = 0$ to w do
(13) for $q = 0$ to h do
(14) $x_1 \leftarrow p - s/2$ {border checking is not shown}
(15) $x_2 \leftarrow p + s/2$
(16) $y_1 \leftarrow q - s/2$
(17) $y_2 \leftarrow q + s/2$
(18) count $\leftarrow (x_2 - x_1) \times (y_2 - y_1)$
(19) sum \leftarrow intImg[x_2, y_2] - intImg[$x_2, y_1 - 1$] - intImg[$x_1 - 1, y_2$] + intImg[$x_1 - 1, y_1 - 1$]
(20) if (in[p, q] \times count) \leq (sum \times (100 - t)/100)
(21) then out[p, q] $\leftarrow 0$
(22) else
(23) out[p, q] $\leftarrow 255$
(24) end if
(25) end for
(26) end for

```

PROCEDURE 2: Adaptive threshold (in, out,  $w$ ,  $h$ ).

TABLE 2: LPD using adaptive thresholding.

| Number of test images | False positives/negatives |                    | Test accuracy | Time  |
|-----------------------|---------------------------|--------------------|---------------|-------|
|                       | Before applying CCA       | After applying CCA |               |       |
| 217                   | 16%                       | 0%                 | 86.64%        | 80 ms |



FIGURE 18: Successful images for LPD using our proposed method under different rotation and illumination.

properly is 29. Therefore, the average detection rate is 98.38% and the computational time is approximately 49 ms. Figure 18 shows some of the test results under different rotation and illumination.

4.5. Performance Comparison of Some Typical LPR Systems with Our Methods for LPD. See Table 3.

## 5. Conclusion

We demonstrated a procedure for LPD algorithms. We used two methods for our LPD system, cascade AdaBoost and adaptive thresholding. Our proposed system is separated into two stages, offline training and online detection, which make our proposed system extremely simple and effective for LPD. In this paper, we demonstrated that such simplicity and effectiveness allow our method to provide better performance than other existing methods. Most of the existing techniques are tremendously complex and are not suitable for real-time applications; however, our proposed algorithm is not complex, thus rendering it suitable for real-time applications.

TABLE 3: Performance comparison of some typical ALPR systems for LPD.

| Methods             | Main procedures for license plate detection                | Database size | Image conditions                                                         | LPD Rate | Processing time | Real time | Plate format     |
|---------------------|------------------------------------------------------------|---------------|--------------------------------------------------------------------------|----------|-----------------|-----------|------------------|
| [10]                | Sliding concentric windows, histogram                      | 40 images     | 640 × 480 pixels (Different distances and weather, road)                 | 82.5%    | —               | —         | Korean plates    |
| [11]                | Vertical edge, edge filtering, and morphological operation | 350 images    | Different distances and weather and road                                 | 95.2%    | —               | —         | Iranian plates   |
| [12]                | Vertical edge detection, unwanted line elimination         | 664 images    | 640 × 480 pixels (various weather conditions, road)                      | 91.65%   | 47.7 ms         | Yes       | Malaysian plates |
| [13]                | Scan line, texture properties, color, and Hough transform  | 332 images    | 867 × 623 pixels (various illumination and different distances and road) | 97.1%    | 0.53 s          | No        | Taiwanese plates |
| Our proposed method | Cascade AdaBoost algorithm and adaptive thresholding       | 1800 images   | 1280 × 720 pixels, various weather conditions and different illumination | 98.38%   | 49 ms           | Yes       | Korean Plates    |

Using our proposed method, experimental results show that the test accuracy is 98.38% with a computational time of 49 ms, which is significantly faster than other existing methods. In regard to our proposed method, practicing and improving its accuracy and practicality are considerations for future work. Moreover, LP character recognition is our principal future work.

### Conflict of Interests

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

### Acknowledgment

This paper was supported by Wonkwang University in 2013.

### References

- [1] H. Bai and C. Liu, "A hybrid license plate extraction method based on edge statistics and morphology," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04)*, vol. 2, pp. 831–834, August 2004.
- [2] S. K. Kim, D. W. Kim, and H. J. Kim, "A recognition of vehicle license plate using a genetic algorithm based segmentation," in *IEEE International Conference of Image Processing (ICIP '96)*, vol. 2, pp. 661–664, 1996.
- [3] S. Kim, D. Kim, Y. Ryu, and G. Kim, "A robust license-plate extraction method under complex image conditions," *International Conference on Pattern Recognition*, vol. 3, pp. 216–219, 2002.
- [4] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. II366–II373, June–July 2004.
- [5] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [6] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2011.
- [7] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, Thomson Learning, 2008.
- [8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. I-511–I-518, 2001.
- [9] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *IEEE International Conference on Image Processing (ICIP '02)*, vol. 1, pp. 900–903, 2002.
- [10] K. Deb, H. Chae, and K. Jo, "Vehicle license plate detection method based on sliding concentric windows and histogram," *Journal of Computers*, vol. 4, no. 8, pp. 771–777, 2009.
- [11] M. Ashoori-Lalimi and S. Ghofrani, "An efficient method for vehicle license plate detection in complex scenes," *Circuits and Systems*, vol. 2, no. 4, pp. 320–325, 2011.
- [12] A. M. Al-Ghaili, S. Mashohor, A. R. Ramli, and A. Ismail, "Vertical edges-based car license plate detection method," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 26–38, 2012.
- [13] J. M. Guo and Y. F. Liu, "License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1417–1424, 2008.

## Research Article

# Performance Improvement Based Authentication Protocol for Intervessel Traffic Service Data Exchange Format Protocol Based on U-Navigation System in WoT Environment

Byunggil Lee<sup>1</sup> and Namje Park<sup>2</sup>

<sup>1</sup> Electronics and Telecommunications Research Institute (ETRI), 218 Gajeong-ro, Yuseong-gu, Daejeon 305-700, Republic of Korea

<sup>2</sup> Department of Computer Education, Teachers College, Jeju National University, 61 Iljudong-ro, Jeju-si, Jeju-do 690-781, Republic of Korea

Correspondence should be addressed to Namje Park; [namjepark@jejunu.ac.kr](mailto:namjepark@jejunu.ac.kr)

Received 15 March 2014; Accepted 4 June 2014; Published 7 August 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 B. Lee and N. Park. 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.

International Association of Lighthouse Authorities (IALA) is developing the standard intersystem VTS exchange format (IVEF) protocol for exchange of navigation and vessel information between VTS systems and between VTS and vessels. VTS (vessel traffic system) is an important marine traffic monitoring system which is designed to improve the safety and efficiency of navigation and the protection of the marine environment. And the demand of Inter-VTS networking has been increased for realization of e-Navigation as shore side collaboration for maritime safety. And IVEF (inter-VTS data exchange format) for inter-VTS network has become a hot research topic of VTS system. Currently, the IVEF developed by the International Association of Lighthouse Authorities (IALA) does not include any highly trusted certification technology for the connectors. The output of standardization is distributed as the IALA recommendation V-145, and the protocol is implemented with an open source. The IVEF open source, however, is the code used to check the functions of standard protocols. It is too slow to be used in the field and requires a large memory. And the vessel traffic information requires high security since it is highly protected by the countries. Therefore, this paper suggests the authentication protocol to increase the security of the VTS systems using the main certification server and IVEF.

## 1. Introduction

The vessel traffic system (VTS) field that is about maritime safety has mostly relied on overseas technology, unlike the shipbuilding industry that has recently retained the leader's position in the global market as a traditional industry. The VTS technology in the maritime safety field consists of maritime IT technology and has desperately required the grafting with the up-to-date IT technology [1, 2].

In the maritime field, the concept of "e-Navigation" for the grafting of the electronic information technology was introduced into Europe, and the popularity of this concept has been rising internationally in recent two to three years. The e-Navigation promoted by IMO is about collecting/integrating/expressing/analyzing the marine data between ships and the land in harmony through the electronic method with the purpose of marine safety/security and

marine environment protection through the improvement of the sailing-related services.

VTS plays the following key roles with the purpose of materializing the e-Navigation in the marine environment: collecting/integrating/analyzing the various data related to marine traffic control and then providing the data to the applicable ships. The VTS Committee (a professional IALA group) has been actively discussing VTS' role, VTS service, and so forth, in order to establish the new concept suitable for the e-Navigation environment [2, 3].

Internationally, there is a trend that VTS has been evolving to vessel traffic management (VTM) recently and VTS' overall concept has expanded as the framework of the methods and services to enhance the following: the safety in the sailable water, the efficiency in security and shipping, and the marine environment protection. In other words, this service architecture has been changing into a new service

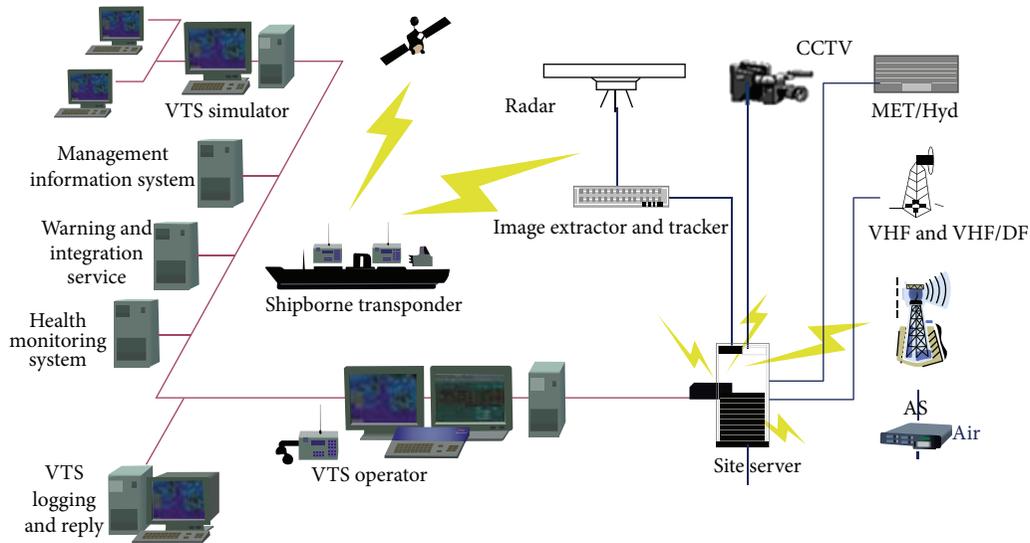


FIGURE 1: System architecture of VTS service.

type, not only in the maritime traffic safety and business service, but also in the maritime computing environment, and foretells the change in the existing VTS and its concept.

In general, the elements composing VTS are shown in Figure 1. VTS is the system in which the followings are connected to one another: the VTS center on the land, the base station site on which various sensors (sensing devices such as CCTV, Radar, DF, MET, etc.) and AIS are installed, the control center that actually operates VTS, and it is a complicated system consisting of various types of telecommunications networks that connect ships, satellites, and sensing devices [4–6].

As for the VTS-related study overseas, through the Framework Programme (FP) project, they promoted various studies whose objects include the next-generation technology of VTS, vessel traffic management and information system (VTMIS), and port control management service (PCS). As the MarNIS project is to be implemented between 2012 and 2020, this study is marked by providing the VTM and search and rescue (SAR) services through collecting various information such as the ship's dynamic/static data and water climate/geography/environment by means of various media and processing the data safely and efficiently. Besides, in the MarNIS project, they have been conducting the aids-to-navigation study (including the marine mobile communication network technology) for the enhanced multimedia telecommunication. In particular, they applied the enhanced controlling function, multimedia telecommunications function, and so forth and have been conducting the follow-up studies and developments continuously for the actual service implementation and the international standardization [7–9].

International Association of Lighthouse Authorities (IALA) is developing the standard intersystem VTS exchange format (IVEF) protocol for exchange of navigation and vessel information between VTS systems and between VTS and vessels. The output of standardization is distributed as the IALA recommendation V-145, and the protocol is

implemented with an open source. The IVEF open source, however, is the code used to check the functions of standard protocols. It is too slow to be used in the field and requires a large memory [10–12].

Secure communication systems among the network enabled devices are significant concern in mobile environments [1].

## 2. Overview of IVEF Protocol

The VTS Committee of IALA is a framework of methods and services to promote the safety, security, efficiency, and environment protection in all transportable water that is evolving from a traditional VTS to an e-Navigation service. In other words, this service structure is advanced from the vessel traffic monitor and control to business service and e-Navigation in vessel computing environment as a new service form. The various information collection and management in the sea encountered a rapid development in its technology, which aims to provide the information service for the vessels during their voyage, such as sea situations and sea map vessel support. At this time, the vessel information collection/management/production/sharing/provision services should be enabled through the information collection from vessels or trusted information exchange between the land systems.

IVEF service is a gateway service in the currently developing land system structure by IALA-AISM's e-Navigation working group. In other words, the IVEF service can have an external third party system linking structure as the client that requests the service and the mutually trusted network gateway security service is required. The traffic information provides the necessary information to the nearby system through the IVEF service. IVEF service should be defined as a mutually linked service between domains. In addition, for a safe IVEF service, the land systems of regional VTS,

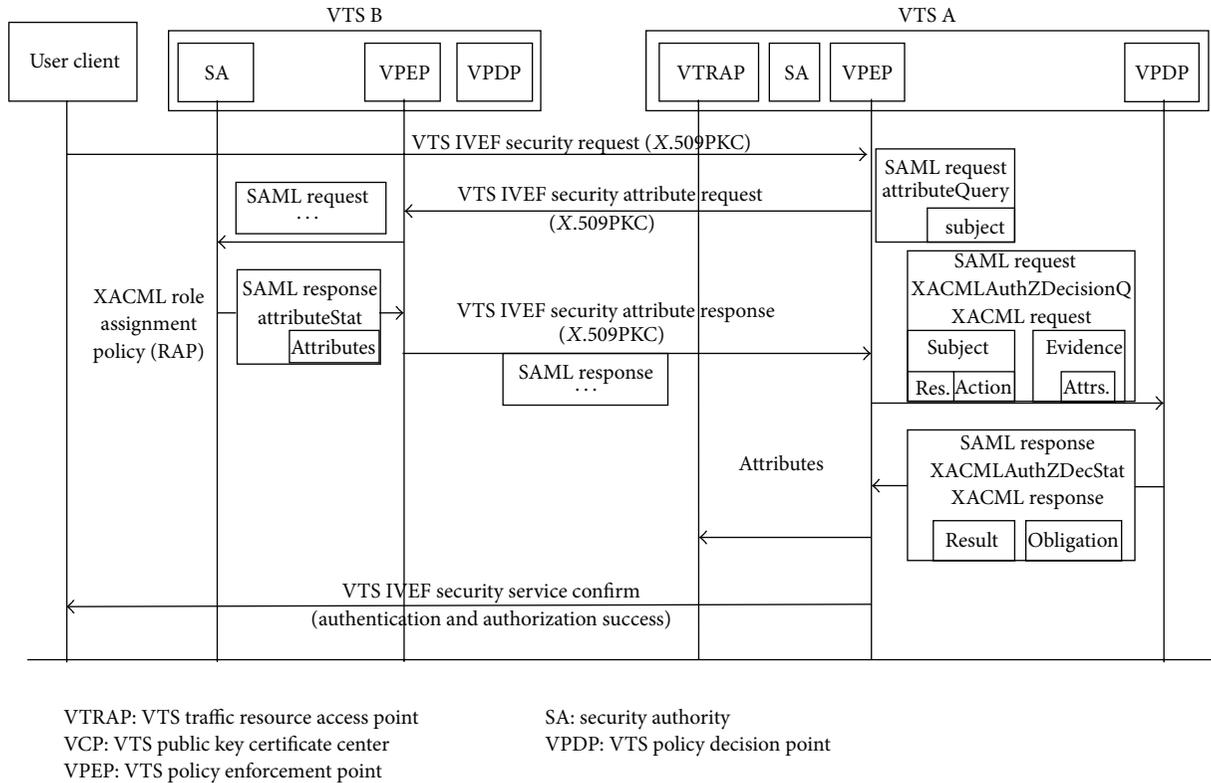


FIGURE 2: IVEF security protocol process.

national VTS, related institutions, and companies should be interconnected in a safe structure.

IVEF service is a server/client model serving as a protocol to exchange traffic information between VTS systems. Its development based on open source is underway by IALA and its protocol and sample program can be checked by downloading SDK in OpenIVEF website [2]. Basic actions to provide service between server/client take three steps as follows. In the first step, a client requests server certification and receives login reply if he/she is a legitimate user. In the second step, the server provides a certain service for the specific user only if it has such service. If it does not offer such service, it provides a basic service defined in the standard called BIS (basic IVEF services). In this step, the client can designate area of interest, data renewal period, or data form based on his/her preference. In the third step, the client sends logout message to the server in order to end use of IVEF service. Since the server does not give a separate reply on the logout message, all the client has to do is just cancel access to server when he/she sends the message [10].

IALA, which is the basic protocol to provide IVEF service between VTS centers, defines nine messages as shown in the Notions and Acronyms section. Definition of these messages is composed of XML-type schema and all messages are composed of subelements of MSG\_IVEF, which is the most significant element. Message of each subelement also has its own sub-elements based on message characteristics. IVEF messages are broadly divided into control information message and real-time information message. The former

consists of user certification and termination, service request to the server and its reply message, and others to provide information on server status. The latter controls ship's current location, expected route, destination port, and other physical information in an object data.

### 3. IVEF Security Process

This clause defines the mutual security factors between domains and detailed procedures using the defined security messages. In other words, Figure 2 shows the security management flow map on the linking areas with the security messages where the VTS domain B approaches VTS domain A. The basic security structure uses the XML based standard protocols and the characteristics for IVEF are expanded using the IVEF security message characteristic exchange protocol. The approach management procedures according to the procedures and authorities for the policy management within a domain when the domains are linked are shown in Figure 4. After the IVEF service between the domains is requested, the VTS IVEF service basic certification mechanism based on ID/Password with the access limitation based authority function is as follows.

- (1) The user sends the access request to use the system resources or application service. At this time, the access request is same as the existing methods with user ID and password.

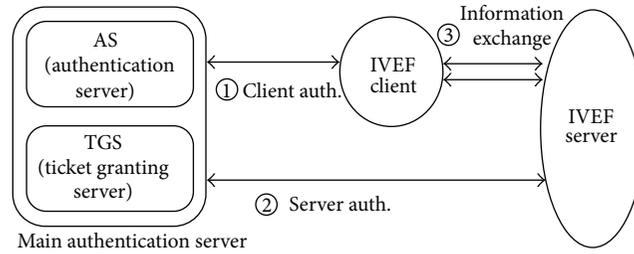


FIGURE 3: Main authorization scheme for user authentication in IVEF.

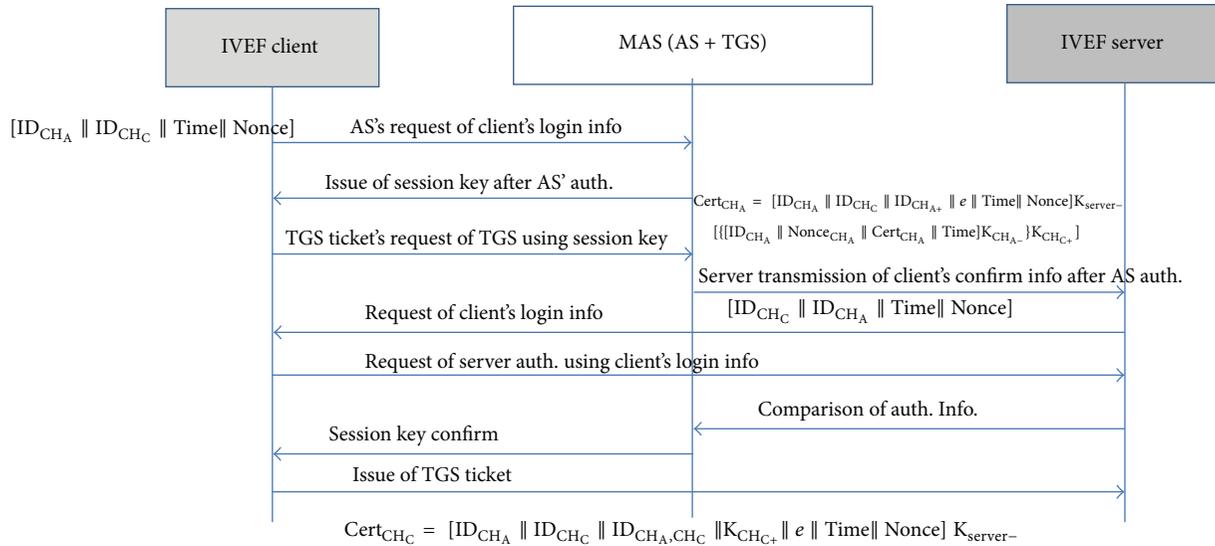


FIGURE 4: Secure protocol between IC, IS, and MAS.

- (2) The PEP of the access control receives the access request and confirms the user's ID and password with the access control list. This is same as the previous method.
- (3) Once the VPEP (VTS policy enforcement point) confirms the user ID and password, it will transmit the user ID and the requested items (read, write, and execute) to VPDP (VTS policy decision point).
- (4) VPDP loads the policy from VPAP (policy administration point) and determines whether the user has the appropriate authorities for the requested actions. For this, the user, resource, environmental characteristics, and policy are used to determine whether to approve.
- (5) VPDP delivers the result to VPEP. In other words, approval/denial is delivered to VPEP. When it is "approved," the user certificate is examined and if it is valid, then the user request is approved.
- (6) VPEP downloads the user certificate from the storage and checks for validity. If it is valid, it approves the access.

#### 4. Security Enhancement of User Authentication Scheme

IVEF is an open-source SDK for VTS information exchange that is being developed by IALA and is almost complete in its international standardization as a gateway. The official IVEF technology documents provided by IALA specify that the data security except for authentication and authorization is out of the IVEF scope. The IVEF security suggested at this point only codes the user authorization information in an open key method. However, when the physical link is terminated and then reconnected between VTSs, the VTS system may be delayed from temporary traffic overload. This may lead to data leakage. A solution requires studies on the main authorization server. This section suggests the main authorization server for user authentication as shown in Figure 3.

Figure 4 briefly summarizes the information exchange system after authentication for the main authentication server with the IVEF client (IC) and IVEF server (IS). The MAS is comprised of AS (authentication server) and TGS (ticket granting server).

Figure 4 shows the protocol between IC, IS, and MAS. Step 1 in Figure 4 shows how IC requires to confirm the user from the AS in MAS using the login information. Step 2 is the

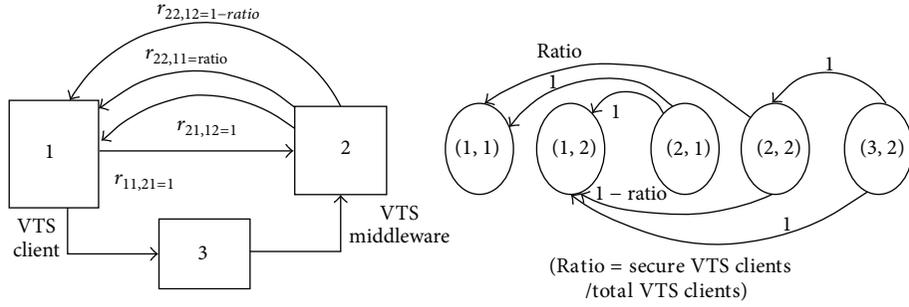


FIGURE 5: Multiple class queuing systems in the secure VTS push scenario.

issuing of the session key after the certification of the IC in AS. Step 3 is the request of the TGS ticket issuing from TGS with the issued session key. TGS ticket holds the client ID, IP address, ticket issue time, and ticket validity information. Step 4 sends the IC confirmation certified in AS in MAS to IS. Step 5 requests the login information from IC by IS directly. Step 6 is the request of the server authentication with the client login information. Step 7 is the result delivery after the comparison of the confirmation from AS in Step 4 and the client login result. Step 8 confirms the issued session key in Step 2. Lastly, Step 9 certifies IC an IS by issuing the same TGS ticket if there were no errors in all steps.

Therefore, MAS can authenticate IC and IS at the same time to sense the link termination in certain areas. In addition, the TGS ticket IP address and ticket valid time information can prevent the illegal access and replay attack of the attackers.

### 5. Security and Performance Discussion of Improved Protocol

We modelled our architecture as a closed queuing system, as in Figure 5, and performed the approximate mean value analysis (MVA) described in [13–15]. In the scenario of Figure 5, the secure mobile VTS procedure has two job classes: the initial secure location update step and secure mobile VTS service step.  $r_{im,jn}$  means the probability that a class  $m$  job moves to class  $n$  at node  $j$  after completing service at node  $i$ . And  $ratio$  represents the ratio of total users to secure mobile VTS service users [15]. The analysis steps for the class switching closed queuing system are as follows.

*Step 1.* Calculate the number of visits in the original network by using

$$e_{ir} = \sum_{j=1}^K \sum_{s=1}^C e_{js} r_{js,ir}, \tag{1}$$

where  $K$  is total number of queues and  $C$  is total number of classes.

*Step 2.* Transform the queuing system to a chain.

*Step 3.* Calculate the number of visits,  $e_{iq}^*$ , for each chain by using

$$e_{iq}^* = \frac{\sum_{r \in \pi_q} e_{ir}}{\sum_{r \in \pi_q} e_{1r}}, \tag{2}$$

where  $r$  is queue number in chain  $q$  and  $q$  is total queue number.

*Step 4.* Calculate the scale factor  $\alpha_{ir}$  and service times  $s_{iq}$  by using (3) with (1):

$$s_{iq} = \sum_{r \in \pi_q} s_{ir} \alpha_{ir}, \quad \alpha_{ir} = \frac{e_{ir}}{\sum_{s \in \pi_q} e_{is}}. \tag{3}$$

*Step 5.* Calculate the performance parameters for each chain using MVA.

Figure 6 showed difference for 4 seconds that compare average transfer time between client and mobile VTS middleware of middleware filtering and unfiltering by network. According as increase tag number on the whole, showed phenomenon that increase until 4 seconds.

Figure 7 showed average transmission time accordingly as increased client number in nonfiltering protocol environment. If client number increases, we can see that average transfer time increases on the whole. And average transfer time which increases rapidly in case of client number is more than 45. Therefore, tag number that can process stably in computer on testbed environment grasped about 40 EA (at the same time). When comparing difference of filtering time and protocol time, time of mobile middleware platform’s filtering module is occupying and shows the importance of signature module about 32% of whole protocol time. In this paper’s supplementary material (see Photos S1 and S2 available online at <http://dx.doi.org/10.1155/2014/734768>), we derive a protocol performance of functions of IVEF client/server.

### 6. Conclusion

These days, the latest electronic technologies and IT are being employed for safer ship operation and efficient control of marine traffic. e-Navigation relies on IVEF service as the standard to exchange data between VTS centers. IVEF,

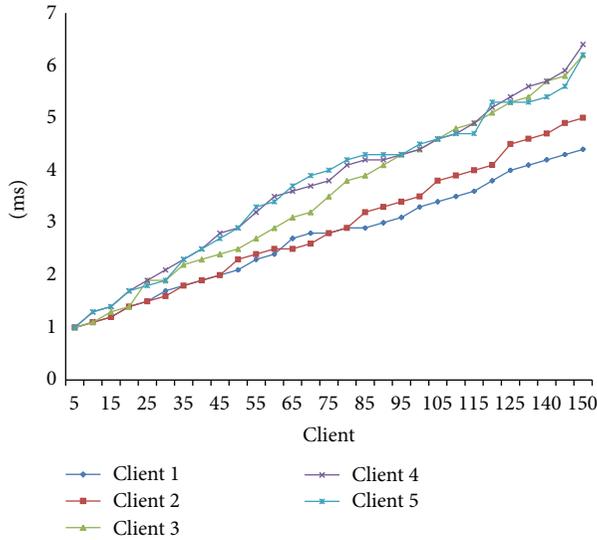


FIGURE 6: Simulation result of mobile VTS middleware filtering.

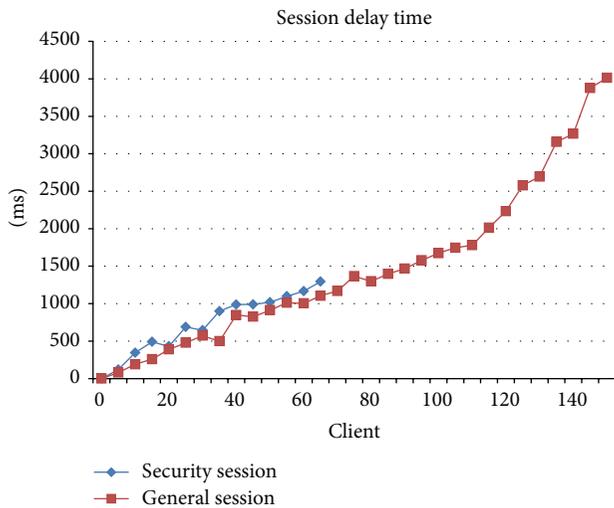


FIGURE 7: Simulation result of mobile VTS middleware nonfiltering.

however, is still a program under development and thus its actual implementation and performance have not been fully verified.

This paper provides an overview of the vulnerability introduced by an attack, as well as the countermeasures to mitigate the threat. Our analysis demonstrates that the Tsai et al.'s protocol does not provide known key security which is a fundamental requirement for secure communication. Our future work is undertaken to protect privacy for mobile stations and improve authentication efficiency.

## Notation and Acronyms

$CH_A$ : Cluster head A  
 $ID_X$ : Identification X

$K_{S,CH}$ : Confidential key shared between session key  $S$  and  $CH$  or  $S$  and  $CH$

Time: Current time

$S$ :  $CH_A$  member client

$X$ :  $CH_B$  member client

$K_{A+}$ : Public key of client A

$K_{A-}$ : Private key of client A

$cert_A$ : Certification of client A

$e$ : Effective date of authentication

$Nonce_A$ : Client A nonce generation.

## Conflict of Interests

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

## Acknowledgments

This paper is extended and improved from accepted paper of CSA 2012 conferences. This work was supported by ETRI through Maritime Safety and Maritime Traffic Management R&D Program of the MOF/KIMST (2009403, development of next generation VTS for maritime safety), and this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2013R1A1A4A01013587).

## References

- [1] IALA Recommendation V-145 on the Inter-VTS Exchange Format (IVEF) Service, June 2011.
- [2] <http://en.wikipedia.org/wiki/E-Navigation>.
- [3] OpenIVEF, <http://openivef.org/>.
- [4] N. Park, S. Cho, B.-D. Kim, B. Lee, and D. Won, "Security enhancement of user authentication scheme using IVEF in vessel traffic service system," in *Computer Science and its Applications*, vol. 203 of *Lecture Notes in Electrical Engineering*, pp. 699–705, 2012.
- [5] K. Kim, B. D. Kim, B. Lee, and N. Park, "Design and implementation of IVEF protocol using wireless communication on Android mobile platform," *Communications in Computer and Information Science*, vol. 339, pp. 94–100, 2012.
- [6] T. Kang and N. Park, "Design of J-VTS middleware based on IVEF protocol," in *Grid and Pervasive Computing*, Lecture Notes in Computer Science, 2013.
- [7] B. Arifin, E. Ross, and Y. Brodsky, "Data security in a ship detection and identification system," in *Proceedings of the 5th International Conference on Recent Advances in Space Technologies (RAST '11)*, pp. 634–636, Istanbul, Turkey, June 2011.
- [8] N. Park and H.-C. Bang, "Implementation of vessel traffic system's mobile middleware platform for secure IVEF service," Security and Communication Networks, 2014.
- [9] D. Frejlichowski and A. Lisaj, "Analysis of lossless radar images compression for navigation in marine traffic and remote transmission," in *Proceeding of the IEEE Radar Conference (RADAR '08)*, pp. 1–4, Rome, Italy, May 2008.
- [10] "A Security Architecture of the inter-VTS System for shore side collaboration of e-Navigation," 2012.

- [11] International Association of Lighthouses and Aids-to-Navigation Authorities (IALA), “Interface Control Document for IVEE” Release 0.1.7.
- [12] <http://en.wikipedia.org/wiki/e-Navigation>.
- [13] N. Park, J. Kwak, S. Kim, D. Won, and H. Kim, “WIPI mobile platform with secure service for mobile RFID network environment,” in *APWeb Workshops 2006*, H. T. Shen, J. Li, M. Li, J. Ni, and W. Wang, Eds., vol. 3842 of *Lecture Notes in Computer Science*, pp. 741–748, Springer, Heidelberg, Germany, 2006.
- [14] N. Park, “Implementation of terminal middleware platform for mobile RFID computing,” *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 8, no. 4, pp. 205–219, 2011.
- [15] Z. C. Taysi and A. G. Yavuz, “ETSI compliant GeoNetworking protocol layer implementation for IVC simulations,” in *Proceedings of the International Conference on Computer, Information and Telecommunication Systems (CITS '12)*, pp. 1–5, Amman, Jordan, May 2012.

## Research Article

# A Rational Threshold Signature Model and Protocol Based on Different Permissions

Bojun Wang,<sup>1</sup> Cheng Cai,<sup>1</sup> and Quan Zhou<sup>2</sup>

<sup>1</sup> School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China

<sup>2</sup> College of Computer Science and Technology, Beijing University of Technology, Beijing 100124, China

Correspondence should be addressed to Cheng Cai; frankgt40@gmail.com

Received 3 April 2014; Revised 1 July 2014; Accepted 4 July 2014; Published 23 July 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Bojun 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.

This paper develops a novel model and protocol used in some specific scenarios, in which the participants of multiple groups with different permissions can finish the signature together. We apply the secret sharing scheme based on difference equation to the private key distribution phase and secret reconstruction phrase of our threshold signature scheme. In addition, our scheme can achieve the signature success because of the punishment strategy of the repeated rational secret sharing. Besides, the bit commitment and verification method used to detect players' cheating behavior acts as a contributing factor to prevent the internal fraud. Using bit commitments, verifiable parameters, and time sequences, this paper constructs a dynamic game model, which has the features of threshold signature management with different permissions, cheat proof, and forward security.

## 1. Introduction

Secret sharing (SS) scheme, first proposed by Shamir [1] in the paper "How to share a secret," is a significant method used for the important information management. There are other SS schemes presented by Blakeley [2] and Asmuth and Bloom [3]. These  $(t, n)$ -threshold schemes above split the secret to  $n$  shares and distribute these shares to  $n$  legal players, meaning that all the players in the secret sharing system have the same permissions. However, in some specific situations, like in a company, managers and employees are supposed to have different authority in the confidential secret management. As a result, all the SS schemes are not suitable to be applied to such scenario. Later, many scholars devoted themselves to the weighted threshold SS schemes, which can solve the above problem. Shamir was concerned with weighted threshold SS in his paper "How to share a secret"—the president of a company has three shares, the vice presidents have two shares, and others have one share. Later, Morillo et al. [4] developed some main properties related to the information ratio, which measures a secret sharing system's security. After that, many researchers used their work to develop weight SS schemes, and some are with

bipartite [5–7]. Chan and Chang [8] developed a new  $(t, n)$ -threshold scheme based on differential equations, which was completely different from the mechanism of weighted SS scheme and shared the same notion with Li [9]. Instead of the traditional weighted threshold SS schemes, which have the symmetrical permissions limitation, they proposed  $(t_1 + t_2, n_1 + n_2)$ -threshold SS scheme that is based on homogeneous constant coefficient linear difference equation. In the scheme, all players are divided into two groups (denoted by  $A, B$ ) with the different secret management authority; just  $t_1$  players from  $A$  and  $t_2$  players from  $B$  can recover the original secret information. For example, a company divides its business secret into  $(n_1 + n_2)$  shares, in which  $n_1$  shares are possessed by  $n_1$  specific employees and  $n_2$  shares are distributed to  $n_2$  managers. Any  $t_1$  employees and  $t_2$  managers can retrieve the business secret.

Threshold signature is based on SS, which was first proposed by Desmedt and Frankel [10] and based on RSA signature mechanism. Shamir [11] introduced the concept of signature authentication based on identity. Paterson and Schuldt [12] presented efficient identity-based signatures in the standard model. In this paper, to illustrate our model, we

adopt Okamoto's signature method [13], which is based on the identification scheme and is provably secure.

Another important issue about the traditional SS scheme is that they are all based on the assumption that every player is either honest or malicious. However, in practice, players are more likely to be selfish, trying to maximize their own utility. Halpern and Teague [14] introduced the notion of rational secret sharing (RSS) in 2004 and presented a randomized protocol for a  $t \geq 3, n > 3$  SS scheme, which can achieve Nash equilibrium after repeated elimination of weakly dominated strategy. Gordon and Katz [15] improved Halpern's protocol to  $t \geq 2, n > 2$  conditions. The mechanism proposed by Maleka et al. [16] is called repeated rational secret sharing (RRSS), in which the distributor needs to do second-time segmentation of the secret shares and made the players share the subshares repeatedly. Maleka's method uses punishment strategies to prevent players from finking, which is different from Halpern and Teague's RSS protocol, in which some rounds of secret sharing are meaningless.

In this paper, we present a rational threshold signature model, in which the participants are divided into two sets with the different permissions. We adopt the SS scheme based on the difference equations to distribute shares and recover the original secrets. In the recover phrase, players exchange their subshares repeatedly based on Maleka's RRSS scheme. In our model, we use several modules to manage the functions, respectively. The parameter sequence generator is used to generate the parameters of the difference equations and parameter distributor is used to distribute the parameters to the participants as their shares. Rounds controller is used to generate the random number of rounds so that the players cannot know when the repeated games will end. Bit commitment module is utilized for the players to commit their own subshares and verify others'. Besides, when a player cheats in a specific round by sending the wrong subshare, the verifiable module can detect it and the protocol will be stopped so that nobody can acquire the secret.

## 2. Relative Works

*2.1. The Model of Li Bin Scholar.* The model is outlined as follows.

Maker constructs homogeneous constant coefficient linear differential equation:

$$a_n + \sum_{i=1}^{t_1} b_i a_{n-i} = 0 \quad (b_i \in Z_q), \quad (1)$$

Master key:  $k = a_N$  ( $N > n_1$ ),

Shadow keys of participants in set  $A$  are  $(a_i, b_1)$  ( $i = 0, 1, \dots, n_1 - 1$ ),

Shadow keys of participants in set  $B$  are  $(N, b_2, \dots, b_{t_1})$ .

The general term formula of homogeneous constant coefficient linear differential equation is

$$a_n = \sum_{i=1}^{t_1} c_i f_i(n). \quad (2)$$

Because coefficient determinant is nondegenerate second-order tensor,

$$\Delta_{t \times t} = \begin{vmatrix} f_1(0) & f_2(0) & \cdots & f_{t_1}(0) \\ f_1(1) & f_2(1) & \cdots & f_{t_1}(1) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(t-1) & f_2(t-1) & \cdots & f_{t_1}(t-1) \end{vmatrix}_{t \times t} \neq 0. \quad (3)$$

Participants in set  $A$  calculate constant vector:

$$c = (c_1, c_2, \dots, c_{t_1})^T. \quad (4)$$

Any participant in set  $B$  makes  $n = N$  can obtain the system master key:

$$a_N = \sum_{i=1}^{t_1} c_i f_i(N) \quad (N > n_1). \quad (5)$$

*2.2. Problems.* The model mentioned above is a big innovation in the field of threshold structure; however, if applied directly to the threshold signature, while in practical use, some problems may exist as follows.

- (1) The permissions in this model have limitations. The second component of  $(n_1 + n_2, t_1 + 1)$ -threshold shared structure on behalf of the second category participants with special privileges; these participants have excessive permissions, because anyone of them can represent the group. Thus, we expand the second component into  $(n_1 + n_2, t_1 + t_2)$  structure. Wei et al.'s scholars [17, 18] at Shandong University have proposed the definition of such structure. However, when this scheme is implemented, its two groups both use the polynomial ring, which possesses the symmetrical nature, thus it will break the different privileges characteristic of the homogeneous constant coefficient linear differential equation. This paper promotes  $(n_1 + n_2, t_1 + 1)$  structure based on homogeneous constant coefficient linear differential equation, extends permissions, in the meantime, and improves the original proposal.
- (2) This model cannot resist conspiracy attacks, because of that when greater than or equal to the  $(t_1, 0)$  threshold number of participants work out the constant vector group of equation (4), at the same time, the equation (2) is determined. Conspirators can get the private key of the participants of the first set, using the general term formula, and one copy of the private key of the second set's participant can be used to conjecture the others' private keys in the second set.
- (3) The model cannot resist internal fraud. When put into practical use, the model does not have a verifiable,

and the participants' fraud is undetectable. If there are no validation measures, the participants may run this protocol arbitrarily, or send their false shares, and these cannot be tolerated.

- (4) The model has the dealer, who is the trusted third party. In the distributed network environment, the parameters is generated by a machine or by the secure multiparty computation.
- (5) This model does not have the rational characteristics. When the signature private keys are generated, and when the first set's participants compute the equation (2)—after computing the general term formula, the participants in the second set have no motive to expose their private key to the participants in the first set, after they generate their private keys. This loses fairness.

### 3. Protocol Model

3.1. *The Structure of Model.* The structure of the model is shown in Figure 1.

(1) *Parameter Sequence Generator.* Each time while in the signature step, the registers in parameters sequence generator dynamically generate the next state parameters according to the last state parameters. Each signature call the module once; the use of time series technology makes the model have forward security.

The initial vector in parameter sequence generator is

$$\begin{aligned} a^{T_0} &= (a_n^{T_0}, a_{n-1}^{T_0}, \dots, a_{n-t_1}^{T_0})^T, \\ b^{T_0} &= (b_1^{T_0}, b_2^{T_0}, \dots, b_{t_1}^{T_0})^T. \end{aligned} \quad (6)$$

The iterative formulas of parameter sequence generator are as follows:

$$\begin{aligned} a^{T_{i+1}} &= (a_n^{\rho T_i}, a_{n-1}^{\rho T_i}, \dots, a_{n-t_1}^{\rho T_i})^T \bmod q \\ & \quad (i \geq 0 \wedge i \in Z^+, \rho \in_R GF(q)^*), \\ b^{T_{i+1}} &= (b_1^{\rho T_i}, b_2^{\rho T_i}, \dots, b_{t_1}^{\rho T_i})^T \bmod q \\ & \quad (i \geq 0 \wedge i \in Z^+, \rho \in_R GF(q)^*). \end{aligned} \quad (7)$$

Other parameters are generated like this way.

**Theorem 1.** *The model has forward security.*

*Proof.* On the completion of the last signature, in next signature step, the parameter sequence generator precompiled the iteration values in registers. After iteration, according to recurrence relations (7), the last data in registers will not exist. That is to say, this time's signature data in registers will cover the last data in them. According to the recurrence relations

(7), if an attacker wants to get last data in registers, he or she must calculate mode square root:

$$\begin{aligned} a_k^{T_i} &= \sqrt[\rho]{a_k^{T_{i+1}}} \bmod q, \\ & \quad (i \geq 0 \wedge i \in Z^+, k = n, \dots, n - t_1 \wedge \rho \in_R GF(q)^*), \\ b_k^{T_i} &= \sqrt[\rho]{b_k^{T_{i+1}}} \bmod q, \\ & \quad (i \geq 0 \wedge i \in Z^+, k = 1, \dots, t_1 \wedge \rho \in_R GF(q)^*). \end{aligned} \quad (8)$$

The mode square root in polynomial time is computationally infeasible, and the mode indices are random; attacker cannot predict. So the model has forward security.  $\square$

(2) *Rounds Controller.* This model, which runs multiple rounds in the signature process, is a limited time repetitions dynamic game. It is vital in the model and controls the operation of the entire process. Here we use the idea of stochastic process [19] to construct model.

**Theorem 2.** *The distribution of round obeys Poisson distribution with parameter  $\lambda$ .*

*Proof.* In the condition of time limited game process, note that the number of deceptions in each round is  $k$ , with the probability satisfying the following formula:

$$\Pr_k(r_0, r) = \Pr\{N(r_0, r) = k\} \quad (k \in Z). \quad (9)$$

Participants' behavior is independent in each round.

Assuming the number of rounds has continuity, that is to say, the process of game is taken as continuous function with time,

$$\begin{aligned} \Pr_1(r, r + \Delta r) &= \Pr\{N(r, r + \Delta r) = 1\} \\ &= \lambda \Delta r + o(\Delta r) \quad (\lambda > 0 \wedge \forall \Delta r \rightarrow 0), \\ \sum_{i=2}^{\infty} \Pr_i(r, r + \Delta r) &= \sum_{i=2}^{\infty} \Pr\{N(r, r + \Delta r) = i\} \\ &= o(\Delta r) \quad (\lambda > 0 \wedge \forall \Delta r \rightarrow 0), \end{aligned} \quad (10)$$

and it satisfies that

$$N(0) = 0 + o(\varepsilon). \quad (11)$$

$\square$

This means that, the probability of cracking the system with  $\varepsilon$  computational advantages can be negligible, when the threshold signature process is not performed. The model satisfies the four conditions mentioned above and meets the definition of Poisson process with  $\lambda$  intensity. That is,

$$N(r) - N(r_0) \sim \pi(\lambda(r - r_0)). \quad (12)$$

**Theorem 3.** *The expectations rounds of this model are  $\lambda$ , each time the model convergence time complexity is  $O(\lambda)$ .*

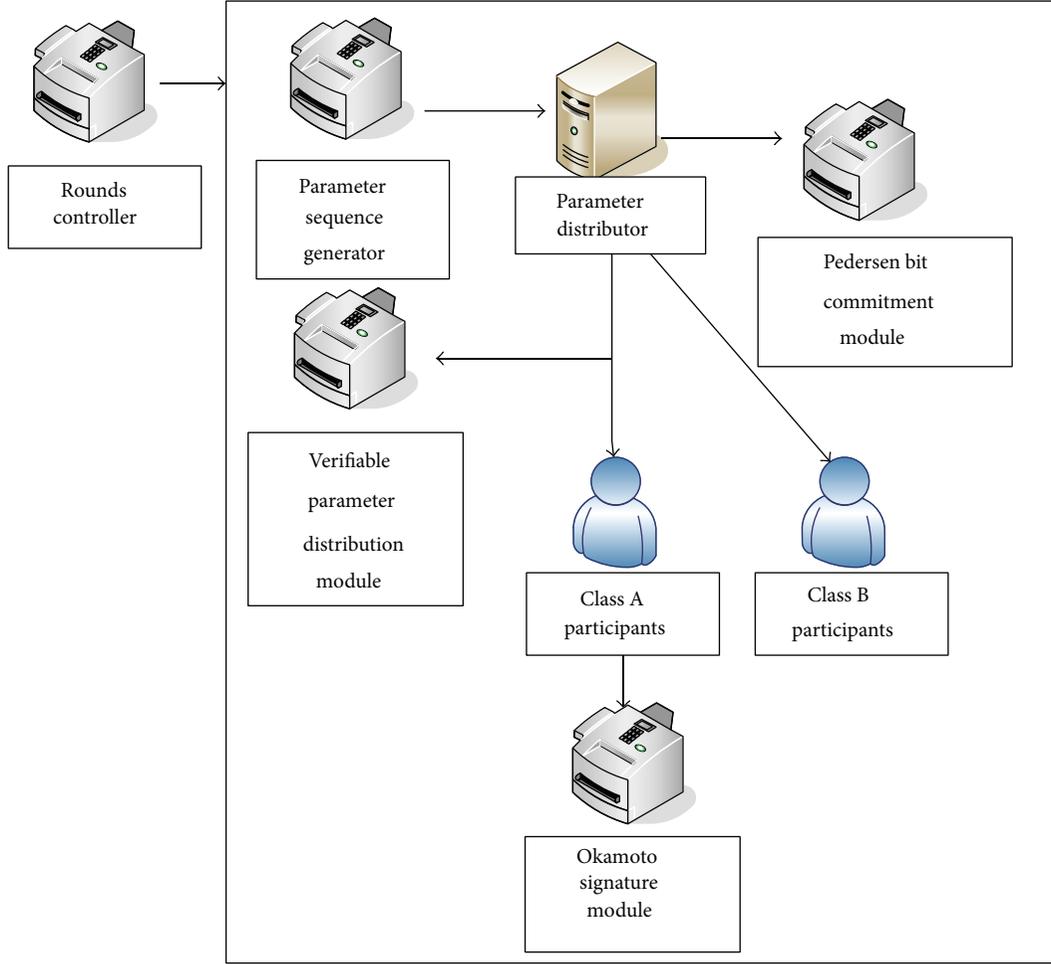


FIGURE 1: The structure diagram of model.

*Proof.* Differential equations are established for the rounds  $(r_0, r_1, \dots, r^*)$  respectively, based on the four conditions mentioned above

$$\begin{aligned} \Pr_k(r_0, r) &= \Pr \{N(r_0, r) = k\} \\ &= \frac{[\lambda(r - r_0)]^k}{k!} e^{-\lambda(r - r_0)} \quad (r, k \in Z). \end{aligned} \quad (13)$$

The mathematical expectation is

$$E[N(r) - N(r_0)] = \lambda(r - r_0). \quad (14)$$

So the expectations rounds of this model are  $\lambda$ , each time the model convergence time complexity is  $O(\lambda)$ .  $\square$

(3) *Parameter Distributor.* A machine can analog the behavior of distributor (maker) and can be a trusted server in the distributed network.

(4) *Pedersen Bit Commitment Module.* Pedersen bit commitment protocol [20] is a security protocol taken as commitment to the bit stream information. In each time of signature,

the system generates coefficients of homogeneous constant coefficients differential equations, and the coefficients of algebraic curved  $F(x)$  with order  $n_2 - 1$ , which correspond to the participants in set B. After storing the coefficients in the binary bits formation, we note them as form of  $m_i$  ( $i \in Z \wedge m_i \in \{0, 1\}$ ), in the form of bits stream. The parameter distributor is also attached with the bit commitment model to prevent it from attacks.

**Theorem 4.** *The model can detect whether the parameter distributor is under attack or not.*

*Proof.* The model adapts the Pedersen's bit stream commitment protocol.

Parameter distributor selects a random number  $\rho \in_R GF(p^{\max\{n_1, n_2\}})^*$ , timestamp information  $t$ , and secure hash function  $H(m_i, t)$  ( $i \in Z \wedge m_i \in \{0, 1\}$ ).

To make bit stream and timestamp above hash process. The primitive element of group  $GF(p^{\max\{n_1, n_2\}})$  is  $g$ ; publish

$$\delta = g^\rho y^{H(m_i, t)} \mod p^{\max\{n_1, n_2\}} \quad (i \in Z \wedge m_i \in \{0, 1\}). \quad (15)$$

The triple  $(\rho, m_i, t)$  will be publish to the public, right after the end of the signature process. Set A and set B participants

can verify commitment to make sure whether parameter distributor is being attacked or not.  $\square$

(5) *Verifiable Parameter Distribution Module.* Using the idea of Feldman's [21] verification. First, publicize bivariate one-way function  $H(x, y)$ . In each threshold signature process, parameter distributor generates polynomial with  $n_1 - 1$  orders which corresponds to set  $A$  participants:

$$G(x) = \sum_{i=1}^{n_1-1} u_i x \text{ mod } p^{n_1}. \quad (16)$$

Our model uses the primitive element in the finite fields  $GF(p^{n_1})$ , which is  $g_1$ , to compute the number of the operation rounds, which is  $r^*$ , according to the Poisson distribution with parameter  $\lambda$ , and then distribute the points sequence:

$$(x_{1i}, y_{1i}) = \left( H\left(s_{1i}, g_1^{r^*}\right), G(x_{1i}) \right) \quad (i = 0, 1, \dots, n_1 - 1). \quad (17)$$

Then it arbitrarily selects  $n_1 - t_1$  points in the field of  $F_{p^{n_1}}(x, y)$  except the ones in the equation (17), and publish them to the public.

Then it saves the vector

$$s_{1i} \quad (i = 0, 1, \dots, 2n_1 - t_1 - 1), \quad (18)$$

and calls Pedersen's bit commitment module.

After that, it broadcasts:

$$V_i = g_1^{u_i} \text{ mod } p^{n_1} \quad (i = 0, 1, \dots, n_1 - 1). \quad (19)$$

Send each participant in set  $A$ :

$$\begin{aligned} \text{TPK}_i &= [a_i + G(0)] \text{ mod } p^{n_1}, \\ (i = 1, 2, \dots, n_1, a_i \in GF(q)^*, G(0) \in GF(p^{n_1})^* > \sup a_i). \end{aligned} \quad (20)$$

In the set  $B$ , the parameter distributor generates the primitive element, which is  $g_2$ , in the infinite field  $GF(p^{n_2})$ , according to this polynomial with  $n_2 - 1$  orders:

$$L(x) = \sum_{i=1}^{n_2-1} l_i x \text{ mod } p^{n_2}. \quad (21)$$

And then, with the rounds number  $r^*$  noted before, the system distributes publish the points sequence:

$$\begin{aligned} (x_{2i}, y_{2i}) &= \left( H\left(s_{2i}, g_2^{r^*}\right), L(x_{2i}) \right) \\ (i = 0, 1, \dots, n_2 - 1). \end{aligned} \quad (22)$$

We adopt  $(n_2, t_2)$  threshold structure constructed by matrix method.  $t_2$  players in set  $B$  participate in the repeated games and recover the secret  $S$  using the published  $n_2 - t_2$  points. As a result, the players in set  $A$  can input  $S$  after they get the general term formula of homogeneous constant coefficient linear differential equation.

Save vector

$$s_{2i} \quad (i = 0, 1, \dots, 2n_2 - t_2 - 1). \quad (23)$$

And call Pedersen's bit commitment module.

After that, it broadcasts:

$$W_i = g_2^{l_i} \text{ mod } p^{n_2} \quad (i = 0, 1, \dots, n_2 - 1). \quad (24)$$

Send each participant in set  $B$ :

$$\begin{aligned} \text{TPK}_j &= [S + L(0)] \text{ mod } p^{n_2}, \\ (j = 1, 2, \dots, n_2, S \in GF(p)^*, L(0) \in GF(p^{n_2})^* > S). \end{aligned} \quad (25)$$

**Theorem 5.** *The model is verifiable.*

*Proof.* When distributing points sequence and broadcasting corresponding authentication information, participants can simultaneously verify the information.

Set  $A$  participants verify

$$g_1^{G(x_{1i})} = \prod_{j=0}^{n_1-1} V_i^{x_{1i}^j} \text{ mod } p^{n_1} \quad (i = 0, 1, \dots, n_1 - 1). \quad (26)$$

Set  $B$  participants verify

$$g_2^{L(x_{2i})} = \prod_{j=0}^{n_2-1} W_i^{x_{2i}^j} \text{ mod } p^{n_2} \quad (i = 0, 1, \dots, n_2 - 1). \quad (27)$$

If the verification succeeds, participants can trust the information sent by others.  $\square$

(6) *Participants.* Participants in two different permissions together constitute the threshold structure  $(n_1 + n_2, t_1 + t_2)$ . In addition,  $|A| = n_1, |B| = n_2$ , and the threshold values are  $|A|_{\text{threshold}} = t_1$  and  $|B|_{\text{threshold}} = t_2$ .

(7) *Okamoto Signature Module.* After calculating the threshold signature private key, take  $\text{TSK} = a_S$  as the first private key component of the signature module, while the second private key component is generated by public key signature method; select private keys; and publicize public keys, respectively. The model adopts Okamoto signature algorithm to signature finally.

**Theorem 6.** *The model can resist conspiracy attack.*

*Proof.* The second component of the private key in Okamoto signature algorithm can avoid conspiracy attacks which are performed by using general term formula to get other participants' private keys when meeting the threshold condition to calculate homogeneous linear differential equations with constant coefficients general term formula in original model. The second component of everyone's private key has to be kept privately by each individual. On condition that the second component of the private key ensures the privacy, the threshold signature cannot be forged. Furthermore, we can establish a mechanism, that is when there is a dispute, the system will check every participant involving the process of signature arise disputes.  $\square$

**3.2. Improved Threshold Model.** We adopt  $(n_2, t_2)$  threshold structure constructed by matrix method.  $t_2$  players in set  $B$  participate in the repeated games and recover the secret  $S$  using the published  $n_2 - t_2$  points. As a result, the players in set  $A$  can input  $S$  after they get the general term formula of homogeneous constant coefficient linear differential equation.

Make two field extensions:

$$\begin{aligned} [GF(p^{n_1}) : GF(q)] &= [GF(p^{n_1}) : GF(p)] [GF(p) : GF(q)], \\ [GF(p^{n_2}) : GF(q)] &= [GF(p^{n_2}) : GF(p)] [GF(p) : GF(q)]. \end{aligned} \quad (28)$$

Expansion order of algebraic number field  $GF(q)$  is

$$\begin{aligned} Q_1 &= [GF(p^{n_1}) : GF(q)] = n_1 * \left[ \frac{p-1}{q} \right], \\ Q_2 &= [GF(p^{n_2}) : GF(q)] = n_2 * \left[ \frac{p-1}{q} \right]. \end{aligned} \quad (29)$$

Remove the noise terms  $L(0)$  and  $G(0)$  to get coefficients information of homogeneous constant coefficient linear differential equation.

### 3.3. Dynamic Game Model

**Definition 7.** The Computable complete and perfect information dynamic game  $\tau = [P, T, A, S, R, H, I, O, U]$  satisfies:

Participants are noted as  $P = \{\text{Simulator}, P_i\}$  (Simulator represents the nature and parameter distributor).

The set of Types is  $T = \{T_i\}$  ( $T_i \in \{\text{honesty, fraud}\}$ ).

Actions set is  $A = \{A_i\}$  ( $A_i \in \{\text{honesty, fraud}\}$ ).

Strategy set is  $S = \{S_i\} \varphi : (T_i, H_i, I_i, A_i) \rightarrow S_i$ .

Rounds set is  $R \in O(\lambda) \wedge R \in Z^+$ .

Full history set  $H = \{h \mid h = \bigoplus_{i=1}^k A_i\}$  ( $i \in R \wedge 0 \leq k \leq R$ ) is depicted as game tree, whose root is empty history node  $\emptyset$ .

The information set  $I = \{I_i\}$  can be tested and is perfect.

Outcome set is  $O = \{O_i\} \gamma : (A_i, S_i) \rightarrow O_i$ .

Utility function set is  $U = \{U_i\} \gamma \circ \varphi : (T_i, H_i, I_i, A_i, S_i, O_i) \rightarrow U_i$  and satisfies  $\partial^2 U_i < 0$ .

The above game  $\tau$  can be calculated in polynomial time.

**Definition 8.** Computable complete and perfect information dynamic game with  $t_1 + t_2$  elastic equilibrium will reach the equilibrium results, under the conditions that it satisfies the Definition 7 and that each participants is rational. That is,  $U(\sigma_i, \sigma_{-i}) < U(\sigma_i^*, \sigma_{-i})$ ,  $\sigma$  is multiple real variable function  $\sigma : (T_i, H_i, I_i, A_i, S_i, O_i, U_i) \rightarrow U(\sigma_i, \sigma_{-i})$ .

**Theorem 9.** The model converges to computable complete and perfect information dynamic game with  $t_1 + t_2$  elastic equilibrium.

*Proof.* Participants who accord with threshold signature conditions possess superiority of  $\text{Pr} = \varepsilon$  ( $0 < \varepsilon < 1$ ). They can get threshold signature private key without the normal operation of the model. Definitions of utility functions are as follows:

$U_{(0,i)}^{++}$ : participants' ideal utility without the normal operation of the model to obtain the threshold signature private key;

$U_{(r,i)}^+$  ( $0 \leq r \leq r^*$ ): the utility that participant  $i$  gets signature private key and others do not get it in  $r$  round;

$U_{(r,i)}^-$  ( $0 \leq r \leq r^*$ ): utility that participant  $i$  does not comply with the normal execution of the model when model run  $r$  round;

$U_{(r,i)}$  ( $0 \leq r \leq r^*$ ): utility that participant  $i$  complies with the normal execution of the model when model run  $r$  round;

$U_{(r^*,i)}$ : normal utility that participant  $i$  always complies with the operation of the model obtains threshold signature private key when model reaches the last one round;

$U_{(r,\text{all})}^-$  ( $0 \leq r \leq r^*$ ): utility that all participants do not obtain the threshold signature private key. Illustrate that there are some participants had deceived cause model abnormal termination.

Utility function satisfies the strong partial:  $U_{(0,i)}^{++} > U_{(r,i)}^+ > U_{(r^*,i)} > U_{(r,\text{all})}^-$ .

Define events as follows.

A: participant uses the advantage of  $\text{Pr} = \varepsilon$  ( $0 < \varepsilon < 1$ ) to crack threshold signature private key.

B: participant implements protocol.

C: participant takes honesty policy in round  $r$ .

D: participant takes fraud policy in round  $r$ .

We denote the utility of departing from the protocol as  $U_{\text{exception}}$  and denote the expected utility as  $E(U_{\text{exception}})$ . We can get the equation as follows.

$$U_{\text{exception}} = \varepsilon U(\text{Pr}(A)) + (1 - \varepsilon) U(\text{Pr}(B)),$$

$$U(\text{Pr}(B))$$

$$= U(\text{Pr}(B | C) \text{Pr}(C) + \text{Pr}(B | D) \text{Pr}(D))$$

$$= \lambda U_{(r^*,i)} + (1 - \lambda) \sum_{i=1}^r U_{(r,i)}^-$$

$$= \lambda U_{(r^*,i)} + (1 - \lambda)$$

$$\begin{aligned}
 & \times \sum_{i=1}^r \left[ \frac{1}{|GF(p)^*||GF(p)^*|} U_{(r,i)}^+ \right. \\
 & \quad \left. + \frac{(GF(p)^* - 1)^2}{|GF(p)^*||GF(p)^*|} U_{(r,\text{all})}^- \right], \\
 U_{\text{exception}} & = \varepsilon U_{(0,i)}^{++} + (1 - \varepsilon) \\
 & \times \left[ U_{(r^*,i)} + (1 - \lambda) \right. \\
 & \quad \left. \times \sum_{i=1}^r \left( \frac{1}{|GF(p)^*||GF(p)^*|} U_{(r,i)}^+ \right. \right. \\
 & \quad \left. \left. + \frac{(GF(p)^* - 1)^2}{|GF(p)^*||GF(p)^*|} U_{(r,\text{all})}^- \right) \right], \\
 [GF(p) : GF(q)] & = \frac{p-1}{q}.
 \end{aligned} \tag{30}$$

In our protocol,

$$U_{\text{exception}} < U_{(r^*,i)}. \tag{31}$$

Distribution function satisfies

$$r^* = \psi(\lambda). \tag{32}$$

The following formulas are met:

$$\begin{aligned}
 \lambda < \Phi * & \left[ \frac{U_{(r^*,i)} - \varepsilon U_{(0,i)}^{++}}{1 - \varepsilon} \right. \\
 & \left. - \sum_{i=1}^r \left( \frac{1}{|GF(p)^*||GF(p)^*|} U_{(r,i)}^+ \right. \right. \\
 & \quad \left. \left. + \frac{(GF(p)^* - 1)^2}{|GF(p)^*||GF(p)^*|} U_{(r,\text{all})}^- \right) \right],
 \end{aligned} \tag{33}$$

in which

$$\begin{aligned}
 \Phi = 1 \times & \left( U_{(r^*,i)} - \sum_{i=1}^r \left( \frac{1}{|GF(p)^*||GF(p)^*|} U_{(r,i)}^+ \right. \right. \\
 & \left. \left. + \frac{(GF(p)^* - 1)^2}{|GF(p)^*||GF(p)^*|} U_{(r,\text{all})}^- \right) \right)^{-1}.
 \end{aligned} \tag{34}$$

The above equation can determine the range of parameters selection, so that the model converges to computable complete and perfect information dynamic game with  $t_1 + t_2$  elastic equilibrium.  $\square$

**Theorem 10.** *The model can resist inner fraud.*

*Proof.* According to Theorem 9, a rational participant will not depart from the protocol execution in any round. The model overcomes the sensitivity of backward induction and adopts mixed strategy equilibrium. If participants adopted a deceptive strategy in the model execution of any round, this caused the decrease in revenue of participants to  $U_{(r,\text{all})}^-$ . When the protocol terminates, punishment strategies can be used, thus putting an end to deceiving behavior effectively. So the model can prevent inner fraud.  $\square$

#### 4. Protocol Procedure

**4.1. Parameters Generation Process.** Determine the order of set  $A$  and set  $B$ ; determine the threshold value according to the requirements, respectively. Select big prime  $q$ ,  $p$  meets  $q \mid (p - 1)$ . Select primitive element  $g_1$  in finite field  $GF(p^{n_1})$  and  $g_2$  in finite field  $GF(p^{n_2})$ . The participants in set  $A$  and set  $B$  select signature private key as the second component of the Okamoto signature, respectively.

Parameter sequence generator generates coefficient constants vector of homogeneous constant coefficient linear differential equation:

$$\begin{aligned}
 a^0 & = (a_1^0, a_2^0, \dots, a_{n-t_1}^0) \quad (a_i^0 \in Z_q), \\
 b^0 & = (b_1^0, b_2^0, \dots, b_{n-t_1}^0) \quad (b_i^0 \in Z_q).
 \end{aligned} \tag{35}$$

Superscript represents signature number of times; 0 represents the first signature.

**4.2. Dynamic Games Process.** Rounds controller according to Poisson distribution with parameter  $\lambda$  secret generates threshold signature round  $r^*$ . According to the number of participants in set  $A$  and set  $B$ , the threshold value generates coefficient constants vector of polynomial  $G(x)$  and  $L(x)$ , respectively:

$$\begin{aligned}
 u^0 & = (u_1^0, u_2^0, \dots, u_{n_1}^0) \quad (u_i^0 \in Z_{q^{n_1}}), \\
 l^0 & = (l_1^0, l_2^0, \dots, l_{n_2}^0) \quad (l_i^0 \in Z_{q^{n_2}}).
 \end{aligned} \tag{36}$$

Superscript signature represents the number of rounds; 0 represents the first round.

Parameter distributor according to (17) and (22) distributes and publicizes points. Participants in set  $A$  and set  $B$  can use the verifiable parameter distribution module for verification. If there is no cheating behavior, the protocol continues to execute. Otherwise, the verifiable parameter distribution module goes to the interrupt processing. In every round of the games, the players in set  $A$  and set  $B$  use the published points sequence and generate  $G(0)^r$  and  $L(0)^r$ , respectively.

TABLE 1: Several models comparison.

| Model                                                    | Verifiable | Bit commitment | Resist conspiracy attack | Forward security | permission | Convergence time                        | Range of parameters                                                                                                                                                                          |
|----------------------------------------------------------|------------|----------------|--------------------------|------------------|------------|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Halpern and Teague                                       | No         | No             | No                       | No               | different  | $O(\frac{5}{\alpha^3})(0 < \alpha < 1)$ | $\frac{\alpha^2}{\alpha^2 + (1 - \alpha)^2} U^+(\sigma_i, \sigma_{-i})$<br>$+ \frac{(1 - \alpha)^2}{\alpha^2 + (1 - \alpha)^2} U^-(\sigma_i, \sigma_{-i})$<br>$> U(\sigma_i^*, \sigma_{-i})$ |
| Gordon and Katz                                          | No         | No             | No                       | No               | different  | $O(\frac{1}{\beta})(0 < \beta < 1)$     | $\beta \leq \frac{U(\sigma_i^*, \sigma_{-i}) - U^-(\sigma_i, \sigma_{-i})}{U^+(\sigma_i, \sigma_{-i}) - U^-(\sigma_i, \sigma_{-i})}$                                                         |
| Computable complete and perfect information dynamic game | Yes        | Yes            | Yes                      | Yes              | different  | $O(\lambda)(\lambda > 0)$               | $\lambda < \frac{U_{(r^*, i)} - \varepsilon U_{(0, i)}^{++}}{(1 - \varepsilon) * U_{(r^*, i)}}$                                                                                              |

Parameter distributor verifies, respectively,

$$[\text{TPK}_i - G^r(0)] \bmod p^{n_1} \stackrel{?}{=} a_i,$$

$$(G^r(0) \in GF(p^{n_1})^* > \sup a_i, i = 1, 2, \dots, n_1),$$

$$[\text{TPK}_j - L^r(0)] \bmod p^{n_2} \stackrel{?}{=} S,$$

$$(S \in GF(p)^* \wedge L^r(0) \in GF(p^{n_2})^* > S, j = 1, 2, \dots, n_2).$$
(37)

If  $r = r^*$  and (37) holds, calculate (2), and then

$$\text{TSK} = a_s \quad (S > n_1 + n_2). \quad (38)$$

If  $r \neq r^*$  and (37) does not hold,  $G(0)^r$  and  $L^r(0)$  equal the expected value and the protocol enters into the next round.

If  $r \neq r^*$  and (37) does not hold, meanwhile,  $G(0)^r$  and  $L^r(0)$  do not equal the expected value, someone of the players have cheated. At this time, the parameter distributor can perceive the cheating behavior so that the player cannot obtain the signature private key. According to Theorem 10, the rational participants will not deceive.

**4.3. Threshold Signature Process.** The Okamoto signature module is used to complete the feature of signature.

Okamoto signature algorithm contains two private keys: the first is threshold signature private key just generated, and the second is each participant's signature private key in set  $A$  and set  $B$ . Only after verification, parameter distributor can call Okamoto signature module. Two private key generation equations are as follows:

$$\begin{aligned} \text{TSK}_1 &= a_s \quad (s > n_1 + n_2), \\ \text{TSK}_2 &= \prod_{i=0}^{t_1+t_2-1} \text{SHA}(m)^{SK_i}. \end{aligned} \quad (39)$$

Verify equation

$$\prod_{i=0}^{t_1+t_2-1} \text{TSK}_2^{PK_i} \stackrel{?}{=} \text{SHA}(m). \quad (40)$$

$m$  is message sequence, and SHA is secure hash function. We use the equation (41) to complete signature.

$$(\sigma_1, \sigma_2, \sigma_3) = \text{Okamoto}(\text{TSK}_1, \text{TSK}_2). \quad (41)$$

Validation process can use standard Okamoto algorithm.

**4.4. Several Models Comparison.** Table 1 is several models comparison. The parameters range of this model uses the limiting form of (31), (32), (33), and (34).

## 5. Conclusion

This paper proposed computable complete and perfect information dynamic game with  $t_1 + t_2$  elastic equilibrium, based on the homogeneous constant coefficient linear differential equation. We constructs a dynamic game model and protocol using time sequences, bit commitments, Feldman's verification method, and Okamoto's signature permissions. The model achieves two different threshold signature permissions. We proved that, during the game, no participant has the tendency of departing from normal operation, so that the model achieves the purpose of preventing fraud. Our method expands the idea of permission and overcomes five inherent problems in homogeneous constant coefficient linear differential equation.

## 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 National Natural Science Foundation of China (No.61170221). The authors appreciate the help as well as the hard work of the editor.

## References

- [1] A. Shamir, "How to share a secret," *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612–613, 1979.
- [2] G. Blakeley, "Safeguarding cryptographic keys," in *Proceedings of the National Computer Conference*, pp. 313–317, AFIPS Press, New York, NY, USA, 1979.
- [3] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 208–210, 1983.
- [4] P. Morillo, C. Padró, G. Sáez, and J. L. Villar, "Weighted threshold secret sharing schemes," *Information Processing Letters*, vol. 70, no. 5, pp. 211–216, 1999.
- [5] C. Padró and G. Sáez, "Secret sharing schemes with bipartite access structure," *IEEE Transactions on Information Theory*, vol. 46, no. 7, pp. 2596–2604, 2000.
- [6] T. Tassa and N. Dyn, "Multipartite secret sharing by bivariate interpolation," *Journal of Cryptology*, vol. 22, no. 2, pp. 227–258, 2009.
- [7] O. Farràs, J. R. Metcalf-Burton, C. Padró, and L. Vázquez, "On the optimization of bipartite secret sharing schemes," *Designs, Codes and Cryptography*, vol. 63, no. 2, pp. 255–271, 2012.
- [8] C.-W. Chan and C.-C. Chang, "A new  $(t, n)$ -threshold scheme based on difference equations," in *Combinatorics, Algorithms, Probabilistic and Experimental Methodologies*, pp. 94–106, Springer, Berlin, Germany, 2007.
- [9] B. Li, "Differential secret sharing scheme based on special access secret sharing scheme," *Journal of Sichuan University (Natural Science)*, vol. 43, no. 1, pp. 78–83, 2006.
- [10] Y. Desmedt and Y. Frankel, "Shared generation of authenticators and signatures," in *Proceedings of Advances in Cryptology-CRYPTO '91, Santa Barbara, Calif, USA, 1991*, pp. 457–469, Springer, Berlin, Germany, 1992.
- [11] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology*, vol. 196 of *Lecture Notes in Computer Science*, pp. 47–53, Springer, Berlin, Germany, 1985.
- [12] K. G. Paterson and J. C. N. Schuldt, "Efficient identity-based signatures secure in the standard model," in *Information Security and Privacy*, vol. 4058 of *Lecture Notes in Computer Science*, pp. 207–222, Springer, Berlin, Germany, 2006.
- [13] T. Okamoto, "Provable secure and practical identification schemes and corresponding signature schemes," in *Advances in Cryptology-CRYPTO '92*, vol. 740 of *Lecture Notes in Computer Science*, pp. 31–53, Springer, Berlin, Germany, 1992.
- [14] J. Halpern and V. Teague, "Rational secret sharing and multiparty computation: extended abstract," in *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC '04)*, pp. 623–632, New York, NY, USA, 2004.
- [15] S. D. Gordon and J. Katz, "Rational secret sharing, revisited," in *Security and Cryptography for Networks*, vol. 4116 of *Lecture Notes in Computer Science*, pp. 229–241, Springer, Berlin, Germany, 2006.
- [16] S. Maleka, A. Shareef, and C. P. Rangan, "The deterministic protocol for rational secret sharing," in *Proceedings of the 22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS '08)*, pp. 1–7, IEEE, April 2008.
- [17] D. Wei and X. Qiuliang, "Special permission-based rational secret sharing scheme," *China Electronic Business: Communications Market*, no. 2, pp. 180–184, 2009.
- [18] W. Dong, *Secret sharing based on game theory and application of the theory [M.S. thesis]*, Shandong University, 2011.
- [19] F. Z. Ben, *Stochastic Process*, Science Press, Beijing, China, 2011.
- [20] Q. Weidong, *Crypto Graphic Protocols Foundation*, Higher Education Press, Beijing, China, 2009.
- [21] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of the 28th IEEE Symposium on Foundations of Computer Science*, pp. 427–437, 1987.

## Research Article

# Development of the Korean Spine Database and Automatic Surface Mesh Intersection Algorithm for Constructing *e*-Spine Simulator

Dongmin Seo, Hanmin Jung, Won-Kyung Sung, and Dukyun Nam

Korea Institute of Science and Technology Information, Daejeon 305-806, Republic of Korea

Correspondence should be addressed to Dukyun Nam; [dynam@kisti.re.kr](mailto:dynam@kisti.re.kr)

Received 20 January 2014; Accepted 6 May 2014; Published 17 July 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Dongmin Seo 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.

By 2026, Korea is expected to surpass the UN's definition of an aged society and reach the level of a superaged society. With an aging population come increased disorders involving the spine. To prevent unnecessary spinal surgery and support scientific diagnosis of spinal disease and systematic prediction of treatment outcomes, we have been developing *e*-Spine, which is a computer simulation model of the human spine. In this paper, we present the Korean spine database and automatic surface mesh intersection algorithm to construct *e*-Spine. To date, the Korean spine database has collected spine data from 77 cadavers and 298 patients. The spine data consists of 2D images from CT, MRI, or X-ray, 3D shapes, geometry data, and property data. The volume and quality of the Korean spine database are now the world's highest ones. In addition, our triangular surface mesh intersection algorithm automatically remeshes the spine-implant intersection model to make it valid for finite element analysis (FEA). This makes it possible to run the FEA using the spine-implant mesh model without any manual effort. Our database and surface mesh intersection algorithm will offer great value and utility in the diagnosis, treatment, and rehabilitation of patients suffering from spinal diseases.

## 1. Introduction

*e*-Health technologies supporting high efficiency and low-cost medical service based on IT-BT convergence technology have grown in importance because medical expense is increasing and many people are asking for customer-driven medical service in today's aging society. In particular, chronic diseases such as degenerative spinal diseases, high blood pressure, diabetes, and cancer cannot be treated adequately and require steady self-care. Therefore, *e*-Health technologies are being widely used to treat chronic diseases widely. Nowadays, next generation chronic disease management technologies are being extensively researched based on mobile service, cloud computing service, social network service, big data analysis service, genome sequencing service, and computer simulation service [1–3]. By 2026, Korea is expected to surpass the UN's definition of an aged society and reach the level of a "superaged society." As a result, degenerative spinal diseases and related surgical procedures will increase exponentially. As of 2007, medical expenses incurred due to

spinal surgery in Korea totaled 178.6 billion won/year, and the treatment duration reached 1.82 million days/year as shown in Figure 1. The resulting medical burden and economic loss are increasing at a rapid rate. Spinal diseases make everyday life of people impossible and impede economic activities, resulting in a compromised quality of life. Between 2002 and 2004, spinal surgery increased at a particularly high rate among the older demographic, by 68.2% among ages 60–69 and by 94.6% among those aged 70 and older. Among the leading causes of hospitalization for ages 65 and older, spinal diseases ranked number 2 with over 65,000 instances [4].

To prevent unnecessary spinal surgery resulting from overtreatment, systematic prediction of treatment outcomes is necessary, including scientific diagnosis, scientific effect analysis, and analysis of spinal rehabilitation exercises. Computer simulations have been utilized in biomechanical research for the past three decades. Today, advances in computer hardware and software are bringing continually increasing simulation accuracy. We have been developing *e*-Spine which is composed of a spine database (DB) and

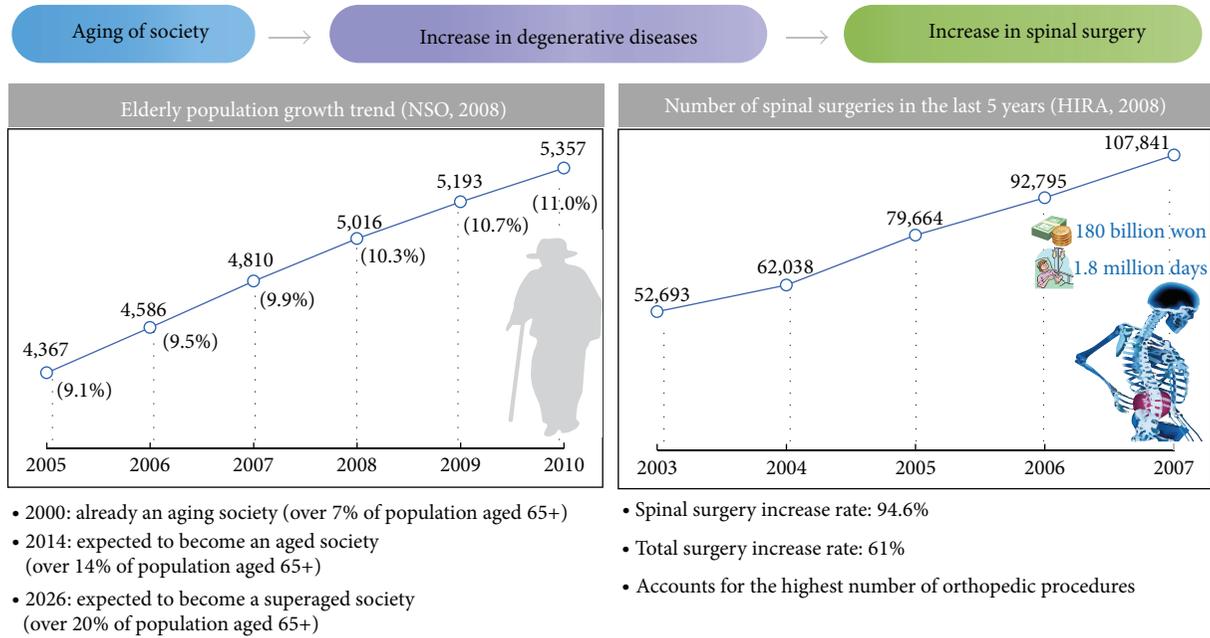


FIGURE 1: Status of an aged society and spinal surgeries in Korea.

a simulation environment for running a computer simulation model of human spines created by mathematically calculating images, geometries, and properties of human spines and will allow virtual testing without using a real human spine. For this, the high quality spine data and a surface mesh intersection algorithm are essential for realizing *e*-Spine (The proposed intersection algorithm is applicable to a triangular surface mesh. In this paper, the surface mesh means the triangular surface mesh). In virtual surgery with an implant, surgeons or biomechanics researchers select the desired model from the spine DB and try to merge the spine model with an implant model as a virtual case. At this point, the two models of the spine and implant are automatically merged by means of the surface mesh merging functionality. With the merged model, they run the computer simulation and analyze the simulation result. To build the spine database (DB) for *e*-Spine, we have produced and collected many images, geometry, and property data of spines from the Korean cadavers and patients with normal spine or degenerative spinal diseases [5, 6]. For automatic mesh merging functionality, we implement the surface mesh intersection algorithm used in the procedure of remeshing the spine-implant intersection model for finite element analysis (FEA). This makes it possible to run the FEA using the spine-implant mesh model without any manual effort.

In this paper, we present our spine database and surface mesh intersection algorithm in detail. The rest of the paper is organized as follows. Section 2 explores *e*-Spine. Section 3 explains the Korean spine database obtained from cadavers and patients with degenerative spinal diseases. Section 4 describes our surface mesh intersection algorithm on the Korean spine data. Section 5 shows our experimental results. Finally, Section 6 presents the conclusion.

## 2. *e*-Spine

*e*-Spine is a computer-run simulation model created by mathematically modeling collected human spinal image data, which allows virtual testing without using a real spine. Figure 2 shows a comparison with a vehicle's navigation system. For optimal and safe driving, a navigation system collects map information, models a map, and predicts a route. Similarly, for optimal treatment, *e*-Spine collects spine images, models 3D spine models, and predicts virtual testing and results. The expected effect of *e*-Spine is as follows:

- (i) acquisition of reliable, economical, advanced IT-based medical support technologies that can be used in the diagnosis and treatment of degenerative spinal diseases;
- (ii) strengthened market competitiveness for Korea's medical equipment industry through the utilization of *e*-Spine;
- (iii) reduction of medical expenses and improving the quality of life during old age by making available reliable, affordable IT-based medical technologies.

## 3. Korean Spine Database

**3.1. Spine Sample Selection.** There are distinct differences, such as facial features, skin color, and hair, between the human races. Differences in the organs of the body are also evident. Generally, spine data obtained from hospitals are only CT, MRI, or X-ray, which focus on particular parts of the spine with degenerative diseases. The data are only available for diagnosing spinal diseases. Therefore, to construct *e*-Spine and support the Korean spine research, we constructed the

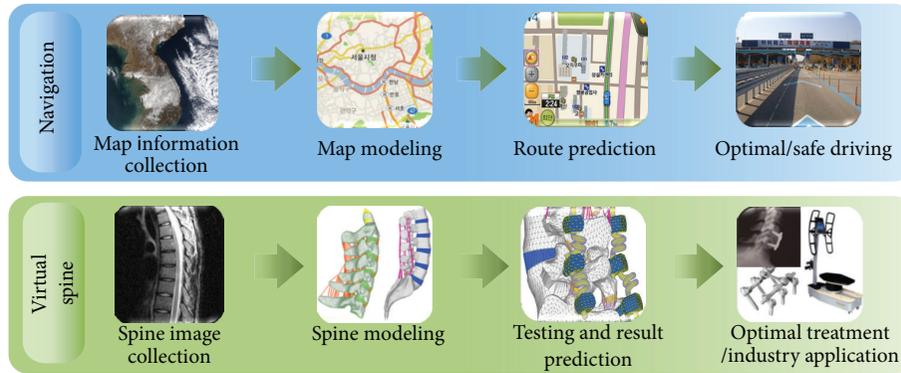
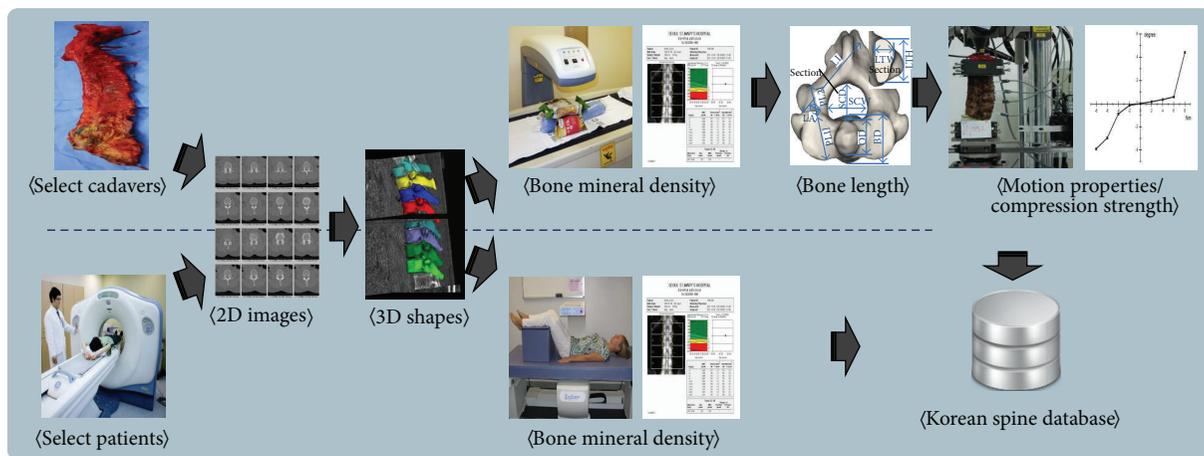
FIGURE 2: Comparison between a vehicle's navigation system and *e-Spine*.

FIGURE 3: Process of the Korean spine database construction.

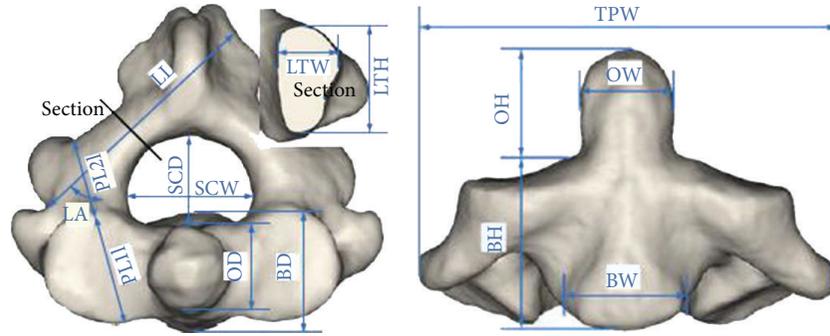
Korean spine data for degenerative spinal diseases. Figure 3 shows the process of the Korean spine database construction. The metadata and schema of our Korean spine database are accepted as a standard from Telecommunications Technology Association and are discussed in detail in [7]. To date, we have collected spine data from 77 cadavers and 298 patients. Among the leading causes of hospitalization for ages 65 and older, spinal diseases ranked number 2 next to cataracts. Therefore, most of the data were obtained from aged cadavers and patients over 50 but some were gathered from younger patients because even young people occasionally suffer from degenerative diseases of their vertebrae.

**3.2. Spine 2D Image and 3D Shape Data.** Our spine 2D images consist of various images from X-ray, CT, MRI, and BMD because different types of images are useful for diagnosing different types of diseases. These images are stored in the Digital Imaging and Communication in Medicine (DICOM) file format, which is a standard format for storing medical data. Some images like those from CT often comprise a series of images produced with small intervals and these images need to be stored and managed as a group in our database for efficient search and management. To accomplish this, we add series numbers to the end of each image file

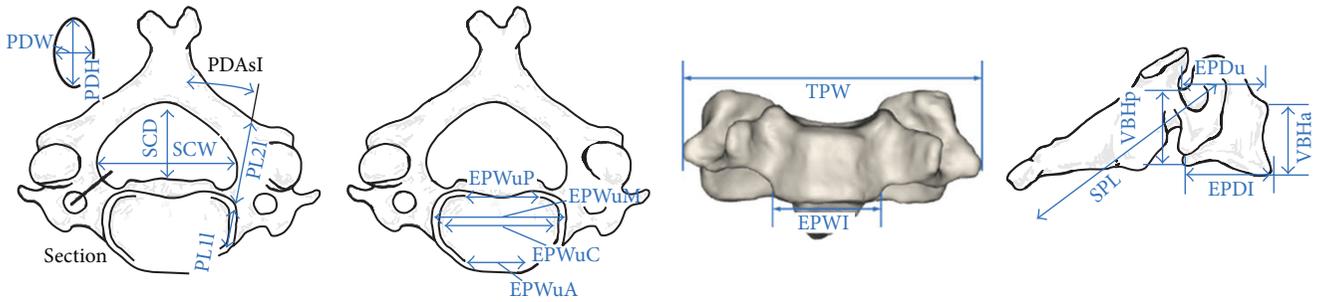
name while sharing the prefix of each file name and manage the file name prefix with start and end of series number as additional metadata. From a series of CT images, we can make a 3D shape model of the spine body by piling up image series in order and filling up small triangles to correct intervals between the images. This process was done with Maya, 3D animation software. Therefore, users can analyze the correlation between cross-sectional CT images and 3D shape models.

**3.3. Spine Geometry and Property Data.** Spine geometry data are lengths and angles of key elements for representing shapes and features of the spine. We selected 481 spine geometry data items in [8–10] and measured those of our cadavers. The selected spine geometry data items are accepted as a standard from Telecommunications Technology Association and are discussed in detail in [11]. We did not measure the geometry data of a patient because we did not harvest the spine from a patient. The selected spine geometry data are utilized to analyze characteristics of the Korean spine. Figure 4 shows the selected elements in the spine geometry data related to cervical vertebrae.

Figure 5 shows the selected elements in the spine geometry data related to thoracic vertebrae.



(a) Cervical vertebra 2



(b) Cervical vertebra 3-7

FIGURE 4: Geometry data related to cervical vertebrae.

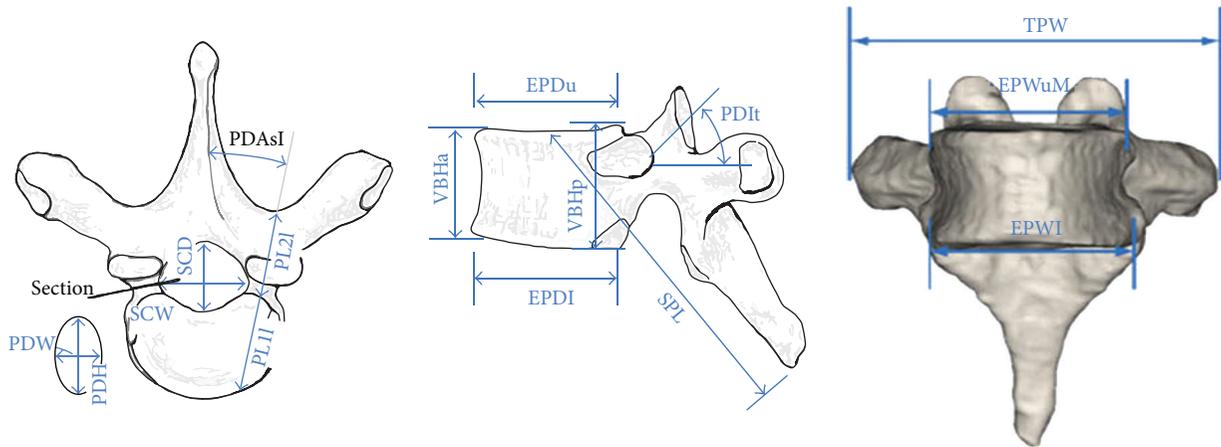


FIGURE 5: Geometry data related to thoracic vertebrae.

Figure 6 shows the selected elements in the spine geometry data related to lumbar vertebrae.

To obtain spine property data, we distribute a spine to cervical vertebrae (C3-C7), thoracic vertebrae (T1-T6, T7-T12), and lumbar vertebrae (L1-L5) and measured flexion-extension, lateral bending, twist, and disc compression between bones. Table 1 shows the test conditions and Figure 7 shows the testing devices consisting of spine simulator of MTS and Liberty of Polhemus for measuring spine property data.

3.4. Degenerative Diseases on Spine. The human spine is divided into three parts: cervical, thoracic, and lumbar

TABLE 1: Test conditions for measuring property data.

| Region   | Maximum moment | Load step | Holding time |
|----------|----------------|-----------|--------------|
| Cervical | 1-2 Nm         | 4         | 30 sec       |
| Thoracic | 4-6 Nm         | 4         | 30 sec       |
| Lumbar   | 8 Nm           | 4         | 30 sec       |

vertebrae, starting from head. As humans age, they often suffer from various degenerative diseases of the cervical and lumbar vertebrae or intervertebral discs, while they rarely suffer from diseases of the thoracic vertebrae. Therefore, we decided to target the following degenerative diseases

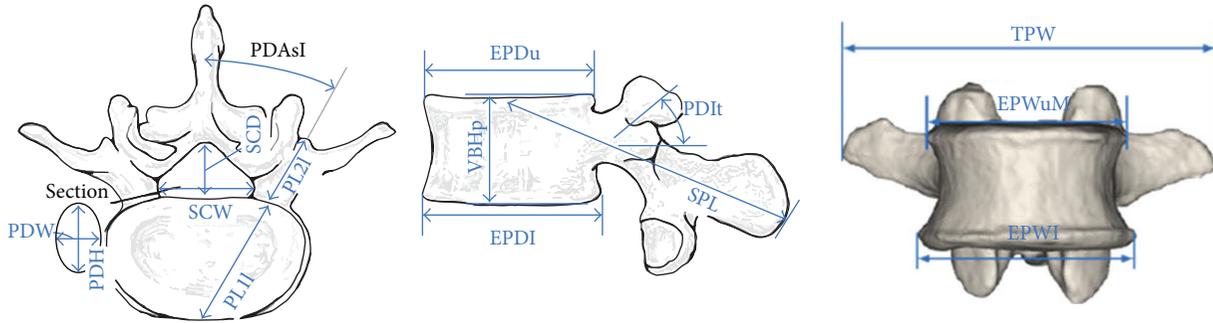


FIGURE 6: Geometry data related to lumbar vertebrae.

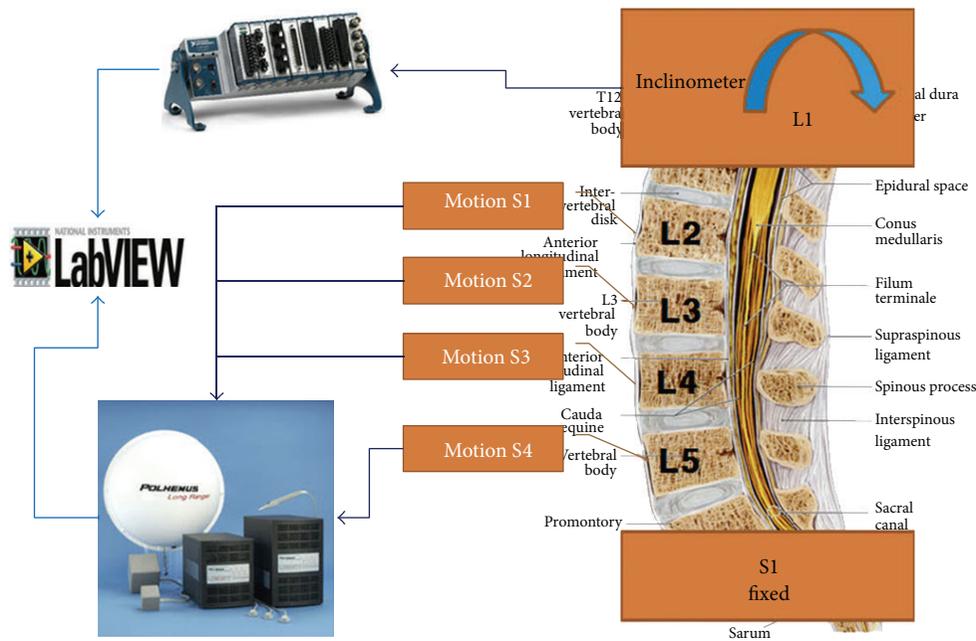


FIGURE 7: Testing devices for measuring property data.

frequently occurring at the cervical or lumbar vertebrae of older people and further decided to classify these diseases into four or five grades (i.e., grade 0 to grade 3 or grade 4) through consulting with several specialists in the human spine. The larger the grade number is, the more severe the disease is. Grade 0 indicates normal and grades 1, 2, 3, and 4 correspond to *mild*, *moderate*, *severe*, and *very severe* grade of disease, respectively. Below is a brief description of the selected degenerative diseases on the spine.

- (i) Osteophyte. This occurs at both cervical and lumbar vertebrae and can be further classified into anterior osteophyte and posterior osteophyte. This can be generally diagnosed by X-ray and CT.
- (ii) Ligament calcification (or ossification). There are three kinds of ligaments around cervical vertebrae: anterior and posterior longitudinal ligaments and ligamentum nuchae. Thus, this disease can be further divided into three subdiseases: ossification of the anterior longitudinal ligament (OALL), ossification

of the posterior longitudinal ligament (OPLL), and ossification of ligamentum nuchae (OLN). This can generally be diagnosed by X-ray and CT.

- (iii) Endplate sclerosis. This occurs at both cervical and lumbar endplates and can generally be diagnosed by X-ray, CT, and MRI.
- (iv) Disc height reduction. This occurs at both cervical and lumbar intervertebral discs and can generally be diagnosed by X-ray and CT.
- (v) Disc herniation. This occurs at both cervical and lumbar intervertebral discs and can generally be diagnosed by X-ray, CT, and MRI. This is usually classified into five grades: normal, bulging, protrusion, extrusion, and sequestration.
- (vi) Disc degeneration. This occurs at both cervical and lumbar intervertebral discs and can generally be diagnosed by X-ray, CT, and MRI. This is also classified into five grades.

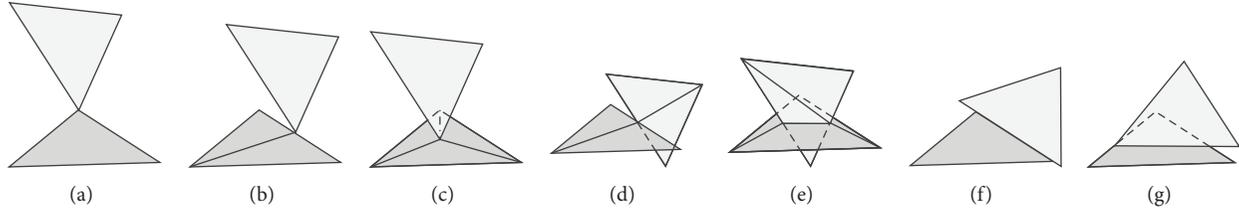


FIGURE 8: Searching for intersection regions. (a) Coincident nodes intersection. (b) Nodes along edge intersection. (c) Nodes onto face intersection. (d) Edges cutting other edge intersections. (e) Edges cutting face intersection. (f) Edges overlapping other edge intersections. (g) Edges overlapping face intersection.

- (vii) Facet joint degeneration. This occurs at both cervical and lumbar facet joints and can generally be diagnosed by X-ray and CT.
- (viii) Spondylolisthesis. This occurs at both cervical and lumbar vertebrae, but most commonly at lumbar vertebrae, and can generally be diagnosed by X-ray.
- (ix) Osteoporosis. This occurs at any vertebrae and can be generally diagnosed by X-ray and BMD (bone mineral density).

3.5. *Statistics about the Korean Spine Data.* To date, we have collected the Korean spine data from 77 cadavers and 298 patients with normal spine or degenerative spinal diseases. The detailed statistics on the collected data is shown in Table 2.

#### 4. Automatic Surface Mesh Intersection Algorithm

To support automatic mesh merging functionality, we implement the mesh intersection algorithm. We applied a tracking algorithm in order to rapidly and accurately explore an intersection. The tracking algorithm finds the intersection regions along the intersection curve from a valid tracking point [12, 13]. After finding an initial intersection point, the algorithm starts at the initial intersection point and creates an intersection curve along the direction to new intersection points. While this happens, the algorithm also searches for intersection regions. We use the existing data structure in [12] and add one factor which can distinguish between spines and implant meshes.

##### 4.1. Finding Intersection Points

4.1.1. *Searching for Intersection Regions.* A plane equation is derived from the outer product of the points of mesh elements. An intersection point is calculated using the topological relation between two intersection planes. After that, we derive the angle from the inner product of three points of a triangle element. If the sum of the derived angles is  $2\pi$ , the intersection point is inside the triangle element. The intersection of planes on three-dimensional space has various cases. When the tracking algorithm searches for an intersection, we determine the intersection from an interrelation of a line and

TABLE 2: The number of cadavers or patients according to degenerative diseases.

| Type              | Vertebra | Disease                  | Number |
|-------------------|----------|--------------------------|--------|
| Cadavers          | Lumbar   | Normal spine             | 22     |
|                   |          | Osteophyte formation     | 50     |
|                   | Cervical | Normal spine             | 23     |
|                   |          | Disc degeneration        | 4      |
|                   |          | Disc height reduction    | 6      |
|                   |          | Disc herniation          | 6      |
|                   |          | Endplate sclerosis       | 6      |
|                   |          | OPLL                     | 23     |
|                   |          | Ossification             | 16     |
|                   |          | Osteophyte               | 10     |
| Patients          | Thoracic | Compression fracture     | 16     |
|                   |          | Compression fracture     | 32     |
|                   | Lumbar   | Disc degeneration        | 20     |
|                   |          | Disc height reduction    | 15     |
|                   |          | Disc herniation          | 17     |
|                   |          | Endplate sclerosis       | 11     |
|                   |          | Facet joint degenerative | 12     |
|                   |          | LDK                      | 20     |
|                   |          | Osteophyte               | 25     |
|                   |          | Osteoporosis             | 20     |
| Spondylolisthesis | 16       |                          |        |

a side and there are cases of intersection between intersected triangles in Figure 8.

4.1.2. *Tolerance.* The intersection cannot be mathematically determined because the numerical calculation of a computer does not work on consecutive space. We have to define a tolerance to calculate an intersection. Suppose that the gap has the difference  $d$  as in Figure 9. If  $d \geq tolerance$ , two triangles are not intersected. Otherwise, they are intersected. When two meshes intersect, the shape of intersected meshes changes. This is because the intersection points are derived from the tolerance by moving existing shapes. The shape of an implant in a spine-implant insertion model should not change using the tolerance. Algorithm 1 defines an intersection generation procedure with the tolerance. In case of an intersection point between edge  $E$  and face  $F$ , the point's position depends on which one is an implant mesh between

```

Input: E (edge), F (surface mesh), tolerance
Output: P (intersection point), p and q (points of E), a , b , and c (points of F)
IF $E \in M_{\text{implant}}, F \in M_{\text{spine}}$
 IF Distance: p to $abc <$ tolerance
 RETURN $P = p$;
 ELSE IF Distance: q to $abc <$ tolerance
 RETURN $P = q$;
 ELSE IF $E \in M_{\text{spine}}, F \in M_{\text{implant}}$
 IF Distance: p to $abc <$ tolerance
 RETURN $P = \text{Create new point } (qp, F)$;
 ELSE IF Distance: q to $abc <$ tolerance
 RETURN $P = \text{Create new point } (pq, F)$;

```

ALGORITHM 1: Tolerance.

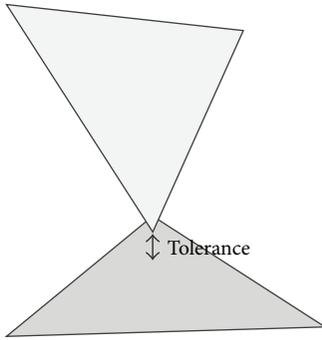


FIGURE 9: Node-face intersection with tolerance.

edge  $E$  and face  $F$ . If  $E$  is an element of an implant mesh, the intersection point is the start or end point of an edge  $E$ . On the other hand, if  $F$  is an element of an implant mesh, the intersection point is generated by intersecting  $F$  and the extension of  $E$ . New mesh points are located on the edge or face of implant mesh because our unique constraint is that implant mesh does not change as much as possible. The tolerance is automatically set up according to the size of a model or by user's configuration.

**4.2. Tracing Algorithm.** In the model for the volume finite element, an intersection curve is always closed, so tracking an intersection can start from an initial intersection point  $P_0$  to the intersection progress direction. After finding an initial intersection point, the intersection points continuously search along the intersection progressing direction [12–14]. An intersection curve is continuously generated by inputting the continuously generated intersection points to an intersection curve  $C$ . Using the intersection curve data, we retrieve intersection regions and generate new mesh of the intersection region.

**4.3. Meshing.** To generate a spine-implant finite element intersection model, the generated mesh on an intersection region should satisfy three conditions.

- (i) The shape of the spine model can be changed but the shape of the implant cannot be changed. This is

because this spine-implant structure analysis simulates the process of inserting a strong durable implant into a relatively weak durable spine. If we allow the shape change of the implant, nonintended stress concentration can occur so the implant shape does not allow the change.

- (ii) We do not generate more meshes than the user needs. The size and the shape of a mesh work are important factors. A small size mesh is densely formed in a complicated shape, but if the number of meshes is blindly large, we waste more analysis resources than necessary.
- (iii) A mesh should have the right quality for the finite element analysis. The intersection point, which can only be created by an intersection search, has an unbalanced gap. A mesh with the intersection points is not enough for finite element analysis, so we need to add fixed points along the intersection curve to create a mesh for finite element analysis.

When generating the intersected mesh model, we have to satisfy the above three requirements. First of all, the points of the intersection curves should be reorganized. Algorithm 2 reorganizes the points of the intersection curve. Figure 10(b) shows the points to define the shape of an implant among the points of an intersection curve.  $d_{\min}$  is defined as the minimum gap of all other points. The unequally distributed points  $P_n$  on the cross-curve are reorganized equally based on the  $d_{\min}$  standard.

The intersection points created on the points or the lines of an implant element are presented as follows:

$$T(P, F_{\text{implant}}) = (F_{\text{implant}}, r, n) \quad (r = 0, 1 \mid n = 0, 1, 2). \quad (1)$$

The algorithm automatically creates a new element network to keep the shape of the original model by using Delaunay triangulation in Figure 10(d). The element created by Delaunay triangulation may be not suitable for finite element analysis. Therefore, the algorithm reorganizes the triangulation element in a spine model through a remesh process. In addition, we remesh the intersection element with neighboring elements to prevent a sharp form. The tracking

```

Input: C (Intersection curve), P (Intersection point)
Output: E_{hard} (Hard edges), V_{hard} (Hard points)
FIND d_{min} (minimum distance between P_k and P_{k+1} ,
 P_k and P_{k+1} are points on the vertexes or edges of F_{implant});
LET P_0 = get first vertex in C ;
FOR $i = 0; i < \text{end of index } P$
 IF $T(P_i, F) = (F_{\text{implant}}, r, n)$ (where $=, 0, 1 \mid n = 0, 1, 2$)
 LET $P_1 = P_i$;
 IF distance between P_0 and $P_1 > d_{\text{max}} \times k$ (user variable)
 FOR $j = 0; \text{distance between } P_{0-j} \text{ and } P_1 > d_{\text{min}} \times k; j++$
 LET P_{0-j+1} = Create new point onto C ;
 LET S_{new} = Create new segment (P_{0-j}, P_{0-j+1});
 INSERT P_{0-j+1} into V_{hard} ;
 INSERT S_{new} into E_{hard} ;
 END
 LET $P_0 = P_1$;
 END
END

```

ALGORITHM 2: Node generation algorithm on the intersection curve to prevent the change of implant model.

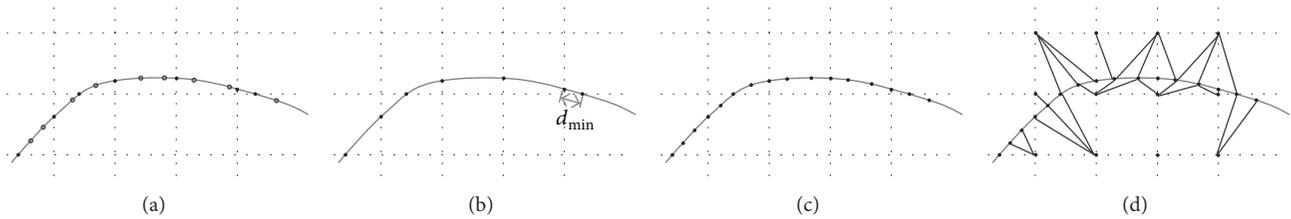


FIGURE 10: Meshing algorithm. (a) Original intersection curve and intersection points (black and white points are the intersection points on implant and spine models, resp.). (b) Intersection points on implant models. (c) Intersection points reconstruction. (d) Triangulation.

algorithm searches for the continuous two points  $P_1$  and  $P_2$  to satisfy this condition. In Figure 11, we generate the points  $P_{1-n}$  with a uniformed gap. The algorithm stores the points on the new generated curve and constructs the line  $S_{\text{new}}$  between newly generated points. The new organized line is sequentially stored as a hard edge,  $E_{\text{hard}}$ . The hard edge is fixed and a base form on mesh or remesh processes.

**4.4. Inserting Implant.** After processing triangulation and remesh, we remove the existing elements of intersection to insert the reorganized elements. The algorithm intersects two finite element models by creating the inserted part inside a target element model. We search for the intersection part with the outer product of surface elements along the intersection curve. In Figure 12, the algorithm removes the region where an implant is inserted in the spine model after searching for the part of the implant inserted in the spine model. It finishes an automatic intersection processing of an implant and a spine by inserting the implant according to the direction to the spine model.

## 5. Performance Evaluation

Figure 13 shows an accurate validation of motion properties with the extension and flexion on L5-S1 on a spine.

The line with the  $x$  symbol is the values of motion properties obtained by [15]. The solid line is obtained by [16]. The line with the squared symbol is obtained by our database. The motion properties from [15] and our database are constructed through experiments. The motion properties from [16] are constructed through computational models. In conclusion, we can confirm that our spine data is accurate because the shape of each line is similar.

A mesh is closely related to an analysis result and the intersection processing in the intersection regions influences the existing shape. Therefore, in most cases, a mesh is manually created to guarantee the mesh quality. We show that the proposed algorithm creates an appropriate mesh for a structural mesh. The algorithm creates a mesh by automatic intersecting of spine and implant models. As the spine model, we use a three-layered spine model consisting of a vertebral arch, an outer vertebral body, and an inner vertebral body. To evaluate the usefulness of the automatic intersection algorithm, we prepare three automatic intersection models with different sizes of implants, as mesh sizes 1.0, 1.5, and 2.0 in Figure 14.

In Figure 15, we set up the analysis conditions: 10,000 load to the  $-Z$  direction on top of the spine with fixed 6 degrees of freedom (DOF). We performed the analysis using an ABAQUS solver.

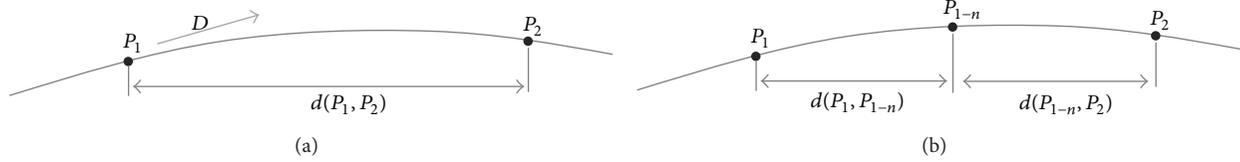


FIGURE 11: Remeshing algorithm. (a)  $P_1$  and  $P_2$  with distance  $d$  on the intersection curve. (b) New generated point  $P_{1-n}$  between  $P_1$  and  $P_2$ .

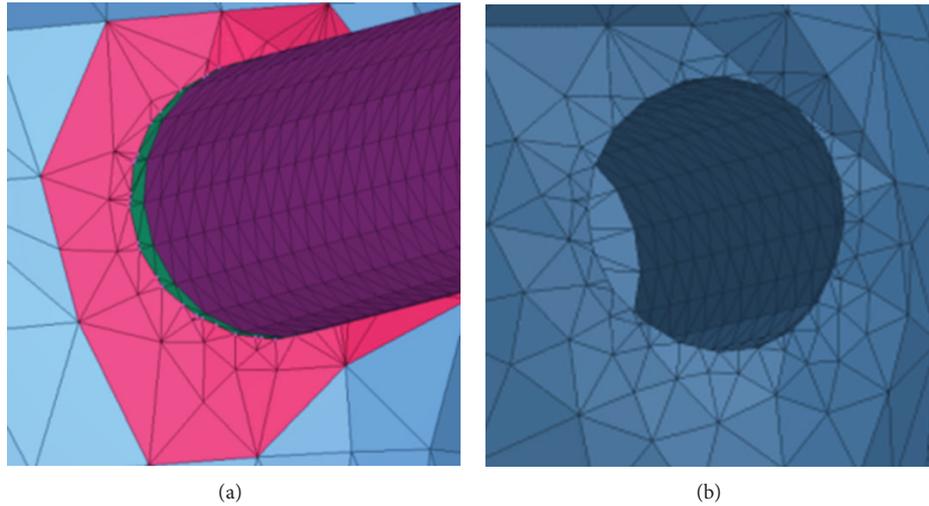


FIGURE 12: Inserting implant. (a) Remeshed intersection model according to the intersection curve. (b) Hole model based on implant shape for automatic intersection.

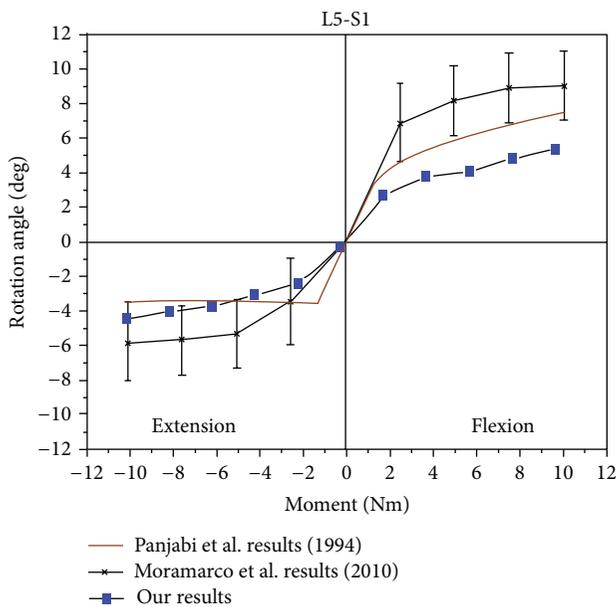


FIGURE 13: Accurate validation of our spine data.

Figure 16 shows the common patterns even though there is little derivation according to the element size. We cannot find the abnormal results due to intersection regions. Therefore, we expect that the proposed automatic intersection algorithm can be used without human intervention.

## 6. Conclusion

We collected various Korean spine data items from 77 cadavers and 298 patients with normal spine or degenerative spinal diseases to provide a wealth of information related to spine to medical students, physicians, and biomedical engineers. We also propose the automatic surface mesh intersection algorithm for spine and implant models. Our algorithm automatically remeshes the spine-implant intersection model to make it valid for finite element analysis (FEA). A spine-implant intersection model is manually created so far. The automatic intersection procedure using the proposed intersection algorithm reduces the manual labor time for spine-implant model. Therefore, this makes it possible to run the FEA using the spine-implant mesh model without any manual effort. We show the validation of intersection mesh quality in the simulation. In the near future, we plan to define criteria to check the quality of a spine-implant intersection mesh and perform the numerical analysis based on the defined criteria.

We will offer our spine data and surface mesh intersection algorithm to many researchers and doctors to foster spine research development. In conclusion, our spine data and surface mesh intersection algorithm will be used to realize reliable, economical, and advanced IT-based medical support technologies that can be used in the diagnosis and treatment of degenerative spinal diseases. Furthermore, our technical skills will be used to vitalize the related research fields and

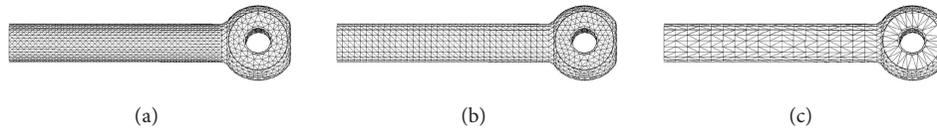


FIGURE 14: Three different implant models. (a) Mesh size 1.0. (b) Size 1.5. (c) Size 2.0.

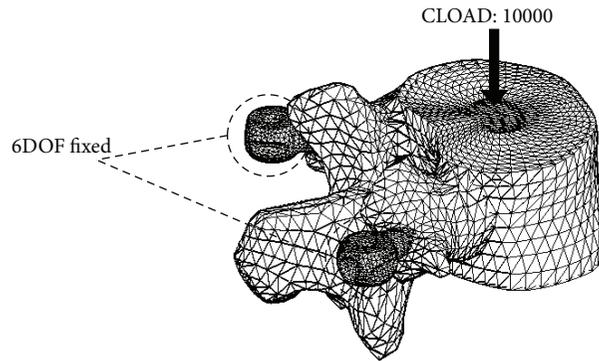


FIGURE 15: Boundary and load condition for evaluation.

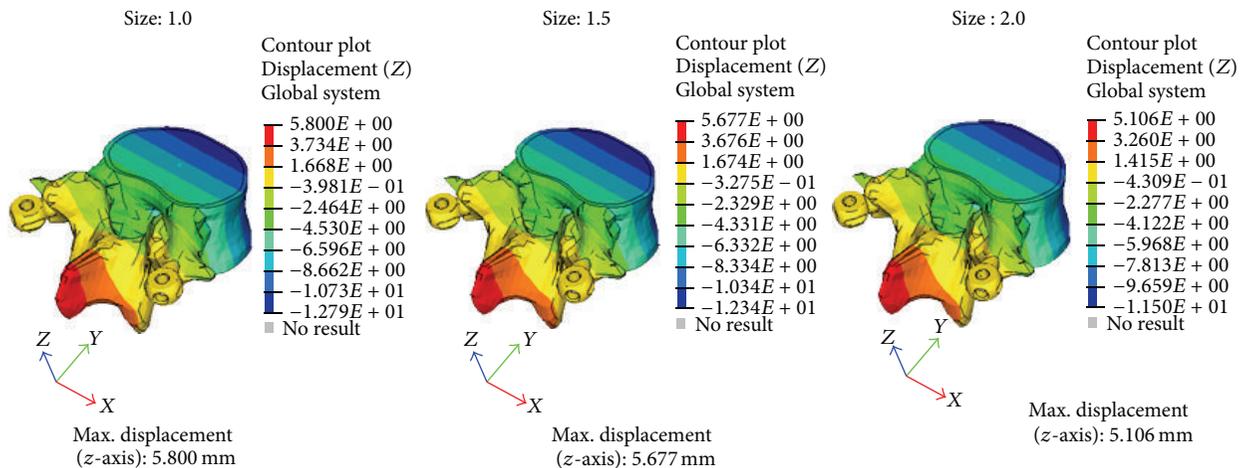


FIGURE 16: Analysis results.

industries by providing the developed human spinal information database, model, and virtual simulator to relevant researchers.

### 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 2014 National Agenda Project (NAP) funded by Korea Research Council of Fundamental Science & Technology (NAP-09-2).

### References

- [1] N. Howard and E. Cambria, "Intention awareness: improving upon situation awareness in human-centric environments," *Human-Centric Computing and Information Sciences*, vol. 3, no. 9, pp. 2–17, 2013.
- [2] P. Gargiulo, T. Helgason, P. Ingvarsson, W. Mayr, H. Kern, and U. Carraro, "Medical image analysis and 3-d modeling to quantify changes and functional restoration in denervated muscle undergoing electrical stimulation treatment," *Human-Centric Computing and Information Sciences*, vol. 2, no. 10, pp. 2–11, 2012.
- [3] M. Brahami, B. Atmani, and N. Matta, "Dynamic knowledge mapping guided by data mining: application on Healthcare," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 1–30, 2013.

- [4] National Health Insurance, "2009 Major surgery statistics," *Press Release*, vol. 12, no. 7, pp. 1–15, 2010.
- [5] "e-Spine project," <http://www.espine.kr/>.
- [6] D. M. Seo, S. W. Lee, S. B. Lee, S. H. Lee, H. M. Jung, and W. K. Sung, "Implementation of spine database system on Korean patient with degenerative spinal diseases," in *Proceedings of International Smart Media Application (SMS '12)*, pp. 1–3, 2012.
- [7] S. B. Lee and S. H. Lee, "Metadata for managing the human body data," Telecommunications Technology Association, TTASstandard (TTAK.KO-10.0723), 2013.
- [8] M. M. Panjabi, J. Duranceau, V. Goel, T. Oxland, and K. Takata, "Cervical human vertebrae: quantitative three-dimensional anatomy of the middle and lower regions," *Spine*, vol. 16, no. 8, pp. 861–869, 1991.
- [9] M. M. Panjabi, K. Takata, V. Goel et al., "Thoracic human vertebrae: quantitative three-dimensional anatomy," *Spine*, vol. 16, no. 8, pp. 888–901, 1991.
- [10] M. M. Panjabi, V. Goel, T. Oxland et al., "Human lumbar vertebrae: quantitative three-dimensional anatomy," *Spine*, vol. 17, no. 3, pp. 299–306, 1992.
- [11] S. B. Lee and S. H. Lee, "The measuring elements for representing the human spine shape," Telecommunications Technology Association, TTASstandard (TTAK.KO-10.0724), 2013.
- [12] A. H. Elsheikh and M. Elsheikh, "A reliable triangular mesh intersection algorithm and its application in geological modelling," *Engineering with Computers*, vol. 30, no. 1, pp. 143–157, 2014.
- [13] S. H. Lo and W. X. Wang, "A fast robust algorithm for the intersection of triangulated surfaces," *Engineering with Computers*, vol. 20, no. 1, pp. 11–21, 2004.
- [14] S. H. Lo, "Automatic mesh generation over intersecting surfaces," *Numerical Methods in Engineering*, vol. 38, no. 6, pp. 943–954, 1995.
- [15] M. M. Panjabi, T. R. Oxland, I. Yamamoto, and J. J. Crisco, "Mechanical behavior of the human lumbar and lumbosacral spine as shown by three-dimensional load-displacement curves," *Journal of Bone and Joint Surgery*, vol. 76, no. 3, pp. 413–424, 1994.
- [16] V. Moramarco, A. P. del Palomar, C. Pappalettere, and M. Doblaré, "An accurate validation of a computational model of a human lumbosacral segment," *Journal of Biomechanics*, vol. 43, no. 2, pp. 334–342, 2010.

## Research Article

# Building a Smooth Medical Service for Operating Room Using RFID Technologies

Lun-Ping Hung,<sup>1</sup> Hsin-Ke Lu,<sup>2</sup> Ching-Sheng Wang,<sup>3</sup> and Ding-Jung Chiang<sup>4</sup>

<sup>1</sup> Department of Information Management, National Taipei University of Nursing and Health Sciences, No. 365, Ming-te Road, Peitou District, Taipei 11219, Taiwan

<sup>2</sup> Department of Information Management, SCE, Chinese Culture University, 55, Hwa-Kang Road, Yang-Ming-Shan, Taipei 11114, Taiwan

<sup>3</sup> Department of Computer Science and Information Engineering, Aletheia University, No. 32, Zhenli St., Danshui Dist., New Taipei City 25103, Taiwan

<sup>4</sup> Department of Computer Digital Multimedia Design, Taipei Chengshih University of Science and Technology, No. 2, Xueyuan Rd., Beitou, Taipei 11112, Taiwan

Correspondence should be addressed to Ching-Sheng Wang; [cswang@mail.au.edu.tw](mailto:cswang@mail.au.edu.tw)

Received 21 January 2014; Accepted 6 May 2014; Published 13 July 2014

Academic Editor: Young-Sik Jeong

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

Due to the information technology advancement, the feasibility for the establishment of mobile medical environments has been strengthened. Using RFID to facilitate the tracing of patients' mobile position in hospital has attracted more attentions from researchers due to the demand on advanced features. Traditionally, the management of surgical treatment is generally manually operated and there is no consistent operating procedure for patients transferring among wards, surgery waiting rooms, operating rooms, and recovery rooms, resulting in panicky and urgent transferring work among departments and, thus, leading to delays and errors. In this paper, we propose a new framework using radio frequency identification (RFID) technology for a mobilized surgical process monitoring system. Through the active tag, an application management system used before, during, and after the surgical processes has been proposed. The concept of signal level matrix, SLM, was proposed to accurately identify patients and dynamically track patients' location. By updating patient's information real-time, the preprocessing time needed for various tasks and incomplete transfers among departments can be reduced, the medical resources can be effectively used, unnecessary medical disputes can be reduced, and more comprehensive health care environment can be provided. The feasibility and effectiveness of our proposed system are demonstrated with a number of experimental results.

## 1. Introduction

The competitiveness of today's medical industry lies not only in the pursuit of higher medical technology breakthroughs, but also in the provision of a patient oriented, safe, and high-quality medical environment as the goal. Surgery has always been one of the busiest tasks in hospitals. Negligence of any details during operation is likely to cause permanent damage on patient. Therefore, it is necessary to strengthen the process of monitoring and management inside operating rooms. How to establish a system that can accurately identify patients, track their locations, and be integrated into the medical information system to enhance the medical service

quality and efficiency has become an important issue of the development of medical science. In this paper, the radio frequency identification, RFID, was imported into the real-time positioning mechanism for surgical patients during the process of surgery. This mechanism can identify patients automatically and detect the locations of the surgical patients accurately from the time they were transferred from the wards into the surgery waiting rooms, operating rooms, and recovery rooms. Signals of RFID devices were transmitted to the back-end server. It is a surgical patients positioning system used before, during, and after the process of surgery. Through the real-time patient information updates to assist medical personnel's medical and nursing operation, they can

have more time to deal with various measures for surgery in attempt to reduce the occurrence of errors and to provide surgical patients with safer medical environments and better medical quality.

The proposed mechanism can overcome the existing blind spot during the process of surgery using wireless transmission technology. RFID technology refers to technology that can identify objects through wireless communication, which is made up of the electronic tag, the reader, and the integration system. Through the communication between the tag and the reader, the reader can access the identification data stored in the tag without contacting with the tag or existing within the visible range of the reader, thereby achieving the function of object identification. From the technical point of view, the tag can be divided into passive mode, active mode, and semiactive mode.

At present, technologies using RFID in the research of positioning are divided into four categories: time of arrival (TOA), time difference of arrival (TDOA), received signal strength (RSS), and angle of arrival (AOA). TOA and TDOA are positioning techniques using time as the basis of measurement and they require accurate synchronization and frequency. TDOA measures the distance by using the relative time of the received signal, whereas TOA uses the absolute time to measure the distance. As compared to the basis of time, RSS uses the signal strength as the basis of measurement, but this method is likely to be interfered by multipath and objects, resulting in measurement errors. AOA uses antenna arrays or directional antennae in LOS (Line-of-sight) signal transmission environment to produce accurate results. RSS is the most suitable positioning mode for indoor environment at present, considering cost and accuracy [1–3].

In this paper, the RSS positioning technology has been adopted. In addition, with the signal strength value that decreases as the distance increases, which is a feature of the RSS, the concept of signal level matrix, SLM, was proposed. Moreover, through the implementation of RFID, patients can be accurately identified and the occurrence of surgical errors can be reduced. Using the real-time patient location positioning system, repeated checking of the processes can be reduced and the surgical processes can be more transparent and automated, thus leading to the establishment of the patient-oriented medical service model. In Section 2, we briefly introduce the development of RFID technology and its application on medical practice. The methodology and the application of the proposed mechanism are described in Section 3. Section 4 contains implementation and experimental results. The last section, Section 5, is conclusions.

## 2. Related Work

Technologies for tracking and tracing are plenty [4–6]. RFID is one of them. It is an important technology for tracking objects. One of the most important RFID applications is positioning and tracking of objects inside building due to its capability of high speed contactless identification. Supply Chain companies like Wal-Mart and Tesco have deployed RFID systems in some of their supply chains. In this paper,

we propose patient position tracing system based on RFID technology. Regarding current development of researches on RFID technology, RFID technologies used in healthcare industry and the position tracing mechanism for surgical procedure. They are described in this section.

*2.1. Current Development of Researches on RFID Technology.* In previous researches, many researches on RFID technology were mainly based on passive mode. Backscattering RFID is a type of passive RFID technology employing tags that do not generate their own signals but reflect the received signals back to the readers [7, 8].

Papers published in the field of RFID research are numerous. Among them, Systems like SotON, LANDMARK, VIRE, and LEASE [9–12] are very famous in the research field of RFID. SpotON uses GPS technology to the RFID indoor positioning system. It uses received signal strength indication to calculate distance and leads to relevant researches in the RFID positioning method. Then, the LANDMARC system was proposed by Liu et al. This system adopts reference tag as the base of improving positioning accuracy. Due to the improvement of positioning accuracy, RFID positioning method has received more attention. Inspired by the LANDMARC system, more refined RFID positioning systems were proposed. Among them, VIRE system uses virtual reference tag and elimination algorithm to reduce the occurrence of errors and to improve the RFID positioning accuracy. LEMT system adds tree structure judgment method to accelerate the calculation of reference tag and area position.

According to previous researches, the main reason causing errors of positioning system is unstable RSSI value of Tag signal, and it is likely to be interfered by environmental factors. Thus, using only signal strength for positioning often results in large errors [13].

*2.2. RFID Technology Used in Healthcare Industry.* RFID is one of the technologies that potentially contribute to the development of medical study. Currently, RFID technology can overcome the bottleneck of workflow management in the field of medical practice. RFID technology can drastically reduce or entirely eliminate the time medical or nurses spend on patient care [14].

Some researches dedicated themselves to the development of point-of-care data delivery based on RFID technology. Staff badges, medication packaging, and patients' identity bracelets contain RFID technology. This facilitates identification of a patient by caregivers, who are, thus, able to submit orders in real-time at the very point of care, instead of being handwritten and sent off for future input. This kind of system saves time and reduces the chances of human error [15, 16]. Moreover, RFID technology can be applied on asset tracking and locating. Some position-based indoor tracking systems have been used in hospitals, where expensive equipment needs to be tracked to avoid being stolen and the patients can get guidance to efficiently use limited medical resources inside complex environments of the hospital. For instance, surgical instruments and other devices must be properly cleaned and packaged between uses.

Tags on the instruments and readers on the sterilization chambers and storage cabinets can validate proper cleaning and help locate needed instruments. RFID can also facilitate better management of medical equipment, medicine, and storage which leads to a more efficient and less medical error environment [17–19].

RFID technology can also be applied on patient location: tracking the location of patients in case of long-term care, mentally challenged patients, and newborns. It has the ability to determine the location of a patient within a hospital and facilitate the delivery of health care. For example, when a patient arrives in a lab for a radiology exam, medical staff is instantly alerted via the RFID tag and the transfer of records can be affected immediately. Patient tracking mechanism can manage patients efficiently to provide a better health care service [20–22].

*2.3. The Positioning Mechanism for Surgical Procedure.* Hospital risk management mostly deals with the high-risk involved activities performed in the hospital. Several RFID technologies are currently available to reduce the risk. The signal strength of RFID chips can provide information to the tagged subject in addition to a unique identity [18, 23]. Active and passive RFID devices were used to track medical personnel, patients, medical equipment, and the deposit of blood to enhance the safety of surgical operation [24]. This adds substantial value to the management of tracing the position of surgical patients. The healthcare environment can be characterized by process broken down into tasks. For instance, a surgery can be broken down into several tasks: signing in at the reception, waiting for the physician, and so forth. Thus, monitoring the surgery progress definitely benefits the adjustment of taking care of personnel, medical equipment, and other related medical resources. Patients can be operated in a safe and efficient medical environment.

Applications of RFID technologies are widely spread these days. Except used in medical industry for personnel identification and management, it is used in operation room for identification and positioning. The passive RFID tracing and positioning monitoring system is proposed in which passive RFID sticker tag is attached to patient's bracelet and medical personnel use PDA to scan RFID bracelet closely to verify patient's identification and location [25]. However, in our opinion, we think that using PDA manually to acquire patient's information stored on RFID tag is still considered manual work which may lead to possible human errors. Active RFID, on the other hand, can automatically detect patient's current location and identification. Standing from the patient's point of view, it is safer this way. Moreover, from the business point of view, issues like saving cost, environmental protection, and long-lasting usage always matter. Using active RFID to replace passive RFID is a better solution for business. In this paper, we proposed signal level matrix, SLM, to improve the accuracy and stability of active RFID indoor positioning system. A detailed description is discussed in later section.

### 3. Methodology

In the operating room, we place RFID reader at the entrance of each passageway to detect and transmit the message to the server when a RFID tag attached hospital bed passing through the reader. Reader refers to the signal strength of a tag to identify its location. However, with the same distance, the performance of the signal strength of tags is quite different that may influence the accuracy of the positioning system. In this paper, to solve the problem of overlapping sensing area and give consideration of the special environment characters of the operating room, a new framework that combines active RFID tags and signal level matrix, SLM, is proposed for the positioning and monitoring of surgical patients in the operating room. Construction of the system is described below.

*3.1. Active RFID Positioning.* In the RFID positioning system, tag and reader communicate wirelessly and the tag has built-in memory; it is divided into active or passive by the source of energy in the tag. The selection of active and passive RFID directly affects the design method, layout, positioning, and correction of the entire system. Therefore, the adoption of the active or passive RFID types is the primary consideration when constructing the system.

The active RFID covers a wider reading range and possesses distinguishable signal strength. The stronger the signal received, the closer the tag is to the reader, and vice versa. Thus, the target objects' whereabouts can be tracked. This feature can be used to calculate the locations of each surgical patient in order to accurately monitor the surgical process. It also aids in extending the preprocessing time for the various undertakings in order to be well prepared for the connection tasks. In addition, the active tag has a bigger memory data space and has the read and rewrite features which are useful in recording the surgical patients' identification. Furthermore, the active RFID can set the time interval for sending tag signals, thus achieving the effect of automatic sensing and reducing the back and forth audit process the medical and nursing personnel go through. Therefore, the active RFID tag is adopted by our proposed system.

*3.2. The Framework of the Proposed Positioning System.* The entire RFID system, including active tags, small power source readers, and an application system, has to be used with the antenna, in which each tag includes a designated number, a model number, place of assembly, and other data. The RFID middleware that we designed can safely send the RFID data read by the reader back to the server-end database. Its structure is as shown in Figure 1. Application program requests information of RFID tags from middleware, middleware via Ethernet requests information of RFID tags from the reader, the reader transmits electromagnetic waves to RFID tags, and then RFID tags passes information back to the reader. Since the tag's designated number is designed in binary data, middleware conduct data processing for received information and then these processed data will be sent back to the application program in the form of XML.

Middleware sets up the relevant communication parameters such as Reader's IP address and Port. After the system issues the connection command, the successfulness of the connecting command is judged by the received data. After successful connection, the data that completed conversion are sent to the buffer to check if the serial number in the buffer is repeated. Data are saved in accordance with the XML document format, signed, and encrypted before being sent back to the server.

Within the changing environment, signal shielding caused by physical properties like reflection, diffraction, and refraction or interference by other electronic signals may influence signal strength which may lead to nonlinear proportional signal strength to distance and misjudgment on determining locations. To overcome the problem of interference, a positioning system usually includes positioning data record in the training database as shown in Figure 1. Based on the result of training database, abnormal signal strength caused by interference can be spotted and adjusted.

The best solution to solve the interferences from indoor static environmental attributes is to reinforce or weaken the signal strength of the tag affected by the interferences. However, for objects in the environment that cannot remain static over a long period of time, such as door opening and closing that instantly interfere with the RSS of the electronic devices, or the shielding effect produced by mobile individuals, we can use the signal weakening feature to judge whether an object exists or use the intuitive method to let the system automatically ignore extreme RSS values.

**3.3. Distribution of Tags and Readers.** Different from current distribution of readers which are mounted on the ceiling, we install reader near the gate and passageway at the position that is the same height with tags attached on hospital bed, shown in Figure 2, to reduce the interference caused by other objects within the hospital.

**3.4. Signal Level Matrix (SLM).** Received signal strength, RSS, is an important data for the RFID system to calculate the location of the tag. The weaker the RSS is, the more distant the range between reference point and target point is; meaning that RFID has the character of diminishing signal accompanied by the increase of distance between reader and tag. When moving a hospital bed with an RFID tag, the reader that is nearer to the tag can be used to locate the position of the tag. If multiple readers detect the existence of the same tag, readers have to determine the tag belonged section. The problem of overlapping sensing area occurred. This paper proposed the signal level matrix (SLM) to solve the multiple detecting problem caused by signal overlapping areas. The signal strength is classified into four levels according to the distance between reader and tag, shown in Table 1. And the signal level matrix of the area around a reader is set up according to the signal levels of the tags received by the reader and its neighboring reader. In Table 1, D is as in detecting radius, meaning that the distance between two readers is 2D meters. In Table 1,  $L0^\dagger$  represents the closest distance between reader and tag;  $L1^\Delta$  represents a longer distance than  $L0^\dagger$ ;  $L2^\#$

TABLE 1: Levels of signal strength.

| Signal level | Distance between reader and tag |
|--------------|---------------------------------|
| $L0^\dagger$ | $<0.5D$                         |
| $L1^\Delta$  | $0.5D\sim 1D$                   |
| $L2^\#$      | $1D\sim 2D$                     |
| $L3^*$       | $>2D$                           |

TABLE 2: The signal level matrix of a tag located in the area of A22.

|              |                    |              |
|--------------|--------------------|--------------|
| $R11 = L3^*$ | $R12 = L3^*$       | $R13 = L3^*$ |
| $R21 = L3^*$ | $R22 = L0^\dagger$ | $R23 = L3^*$ |
| $R31 = L3^*$ | $R32 = L3^*$       | $R33 = L3^*$ |

represents the distance that is longer than  $L1^\Delta$ ;  $L3^*$  represents the longest distance between reader and tag.

As shown in Figure 3, the entire area is divided into 9 sections. One Active RFID reader is located at the center of each section. The distance between two horizontally/vertically neighboring readers is 2D meters. Each reader is labeled with different number ( $R11, R12, \dots$ , and  $R33$ ). There are two sensing ranges for each reader and their radius is D meters and 2D meters, respectively. Intersections occurred within the sensing range of the radius of 2D meters are illustrated as in Figure 3. To distinguish the location of tags, each section is further divided into subsections based on the distance between tag and reader. When a tag is very close to a reader within a range of 0.5D meter, this tag can be easily identified. When the distance between a tag and a reader is larger than 0.5D meter and less than 1 meter, this tag may appear in the ranges of A22U, A22D, A22L, A22R, A22UL, A22UR, A22DL, and A22DR. Noted that D is as in down; U is as in up; L is as in left; and R is as in right. The position of the tag can be identified according to the relative distance between this tag and other reader close by.

Basically, based on the signal levels received by readers, matrix of signal levels received by readers can define a tag's location. If the signal level matrix is as shown in Table 2, this tag is definitely in the range of A22 because R22 receives the strongest signal coming from the tag. If the signal level matrix is as shown in Table 3, a tag's location is determined by the relative signal level received by neighboring readers. Each reader has 8 neighbors. If a tag is closer to one neighbor, that neighboring reader should receive a stronger signal than other neighboring readers. For example, as shown in Table 3(a), a neighboring reader of R22 named R12 receives a stronger signal,  $L2$ , than signal received by other neighbors,  $R11, R13, R21, R23, R31, R32$ , and  $R33$ ; this tag is in the area of A22U. In Table 3, U means upper, D means down, L means left, and R means right. For instance, A22U means upper side of tag number 22.

## 4. Simulation Model and Experimental Results

**4.1. A Simulation Model of the Proposed Positioning System.** Surgery room has always been a busy place. To match the high standard required by the hospital, a simulation model was

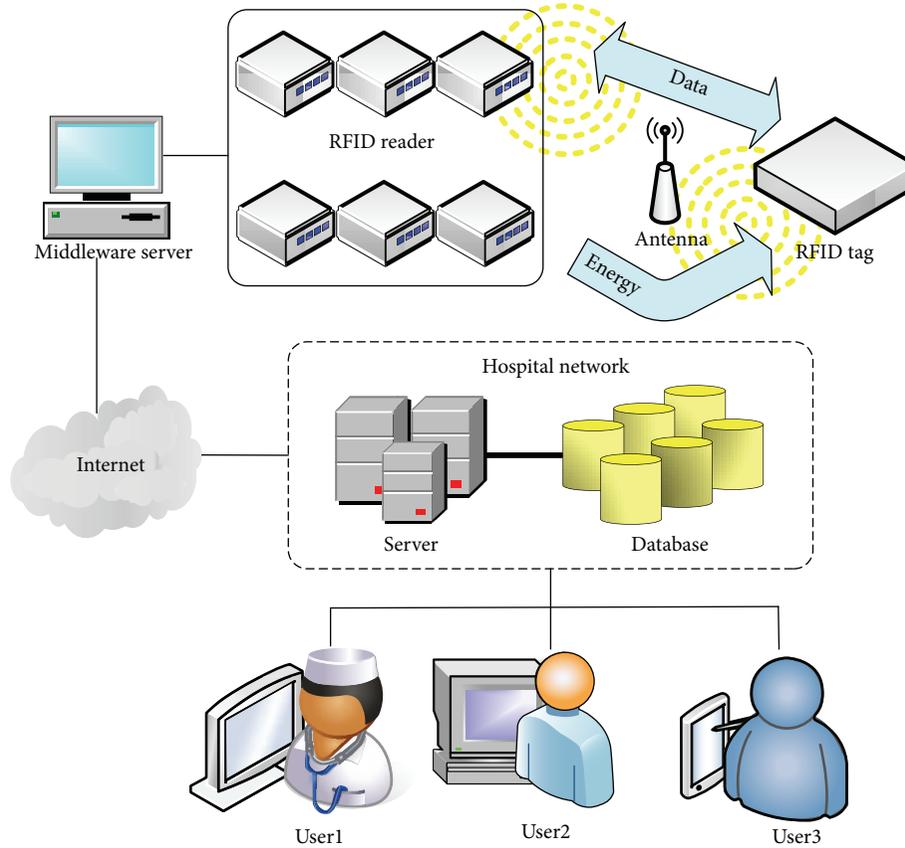


FIGURE 1: Functional component of the RFID System using in medical environment.

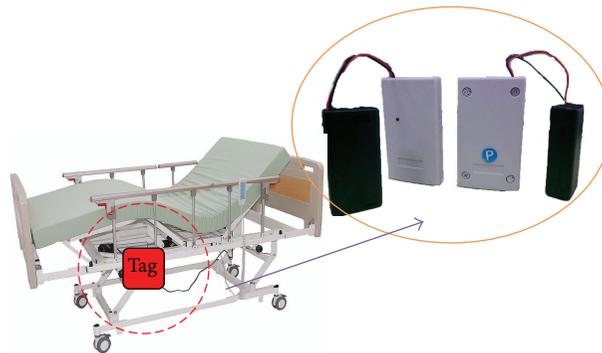


FIGURE 2: The placement of a tag on a hospital bed.

set up in a teaching hospital located in Taipei. As shown in Figure 4, the passageway is formed in U shape in the surgery room based on the layout of the surgery room.

We use MRK-UMR40A RFID reader, UBM-T4ACT RFID tag, and MRK-UMP40A RFID repeater in the surgery room. A RFID repeater is used to transmit the ID number and RSSI to readers. The longest effective transmission distance between a repeater and a reader is 30 meters and the same distance between a repeater and a tag is 20 meters.

Repeater tag will send the tag signal strength received to the reader, so we can regard the reader as the server to receive

the repeater's returned signal. After experiencing a number of practical measurements, we adjusted the signal strength of the tag and the distance between readers, to minimize the influence of signal interference.

Although the RSSI value of a RFID is not very stable, it is reversely proportional to the distance between a reader and a tag. To locate the position of hospital beds, the system can distinguish whether the hospital bed is inside or outside the surgery room by analyzing RSSI due to the segmentation effect caused by wall or door. Furthermore, due to the fact that the nature of the movement of hospital bed is continuous on

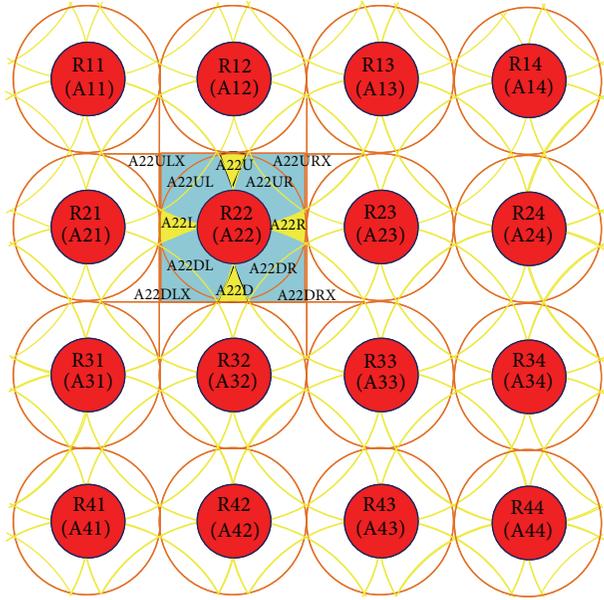


FIGURE 3: The schematic diagram of partitioned sections.

axis, the position of hospital bed can be determined based on the result of SLM.

**4.2. Experimental Result.** An experiment was conducted to determine the installation of the most cost effective RFID environment. In order to simulate the environment of surgery room, we use a  $8\text{ m} \times 6\text{ m}$  lab room and a  $30\text{ m} \times 2\text{ m}$  aisle outside the lab room. In spite of placing a RFID reader inside the lab room, several RFID repeaters were placed along the aisle outside the lab to transmit tag's information like ID number and RSSI values to the reader inside the lab room. Signals received by the reader are shown in Figure 5. There are three tags in this test. Tag 1 is the tag located inside the lab room. Tag 2 and tag 3 are located on the aisle. Among the signal strength sent by these three tags, signal sent by tag 1 can be received stably and its RF value is around  $-50$ . Noted that the range between tag 1 and the reader is around 1.5 meter, the distance between tag 2 and the reader is around 7 meters, and the distance between tag 3 and the reader is around 8 meters. Comparing with signal sent by the tag inside the lab room, signals sent by tags outside the lab room are relatively weak due to the interference caused by door or wall. Therefore, a threshold value is set to be  $-70$  considering the factor of unstable transmission of signals caused by interference.

Besides detecting the signal strength received by the reader inside and outside the surgery room, we use a  $30\text{ m} \times 2\text{ m}$  aisle to simulate the environment outside the surgery room. Figure 6 shows the signal strength sent from a tag to the reader under different distance. The signal strength was measured at least one minute in each experiment. Although signal strength is not very stable during the experiment, basically, signal strength is negatively related to the distance between the reader and the tag within the range of 6 meters. Signal strength fluctuates severely when the distance between the reader and the tag is more than 6 meters. Therefore, signal

TABLE 3: The signal level matrix of a tag located in the areas of A22U, A22D, A22L, A22R, A22UL, A22UR, A22DL, and A22DR.

|                                  |                   |               |
|----------------------------------|-------------------|---------------|
| (a) Signal level matrix of A22U  |                   |               |
| $R11 = L3^*$                     | $R12 = L2^\#$     | $R13 = L3^*$  |
| $R21 = L3^*$                     | $R22 = L1^\Delta$ | $R23 = L3^*$  |
| $R31 = L3^*$                     | $R32 = L3^*$      | $R33 = L3^*$  |
| (b) Signal level matrix of A22D  |                   |               |
| $R11 = L3^*$                     | $R12 = L3^*$      | $R13 = L3^*$  |
| $R21 = L3^*$                     | $R22 = L1^\Delta$ | $R23 = L3^*$  |
| $R31 = L3^*$                     | $R32 = L2^\#$     | $R33 = L3^*$  |
| (c) Signal level matrix of A22L  |                   |               |
| $R11 = L3^*$                     | $R12 = L3^*$      | $R13 = L3^*$  |
| $R21 = L2^\#$                    | $R22 = L1^\Delta$ | $R23 = L3^*$  |
| $R31 = L3^*$                     | $R32 = L3^*$      | $R33 = L3^*$  |
| (d) Signal level matrix of A22R  |                   |               |
| $R11 = L3^*$                     | $R12 = L3^*$      | $R13 = L3^*$  |
| $R21 = L3^*$                     | $R22 = L1^\Delta$ | $R23 = L2^\#$ |
| $R31 = L3^*$                     | $R32 = L3^*$      | $R33 = L3^*$  |
| (e) Signal level matrix of A22UL |                   |               |
| $R11 = L3^*$                     | $R12 = L2^\#$     | $R13 = L3^*$  |
| $R21 = L2^\#$                    | $R22 = L1^\Delta$ | $R23 = L3^*$  |
| $R31 = L3^*$                     | $R32 = L3^*$      | $R33 = L3^*$  |
| (f) Signal level matrix of A22UR |                   |               |
| $R11 = L3^*$                     | $R12 = L2^\#$     | $R13 = L3^*$  |
| $R21 = L3^*$                     | $R22 = L1^\Delta$ | $R23 = L2^\#$ |
| $R31 = L3^*$                     | $R32 = L3^*$      | $R33 = L3^*$  |
| (g) Signal level matrix of A22DL |                   |               |
| $R11 = L3^*$                     | $R12 = L3^*$      | $R13 = L3^*$  |
| $R21 = L2^\#$                    | $R22 = L1^\Delta$ | $R23 = L3^*$  |
| $R31 = L3^*$                     | $R32 = L2^\#$     | $R33 = L3^*$  |
| (h) Signal level matrix of A22DR |                   |               |
| $R11 = L3^*$                     | $R12 = L3^*$      | $R13 = L3^*$  |
| $R21 = L3^*$                     | $R22 = L1^\Delta$ | $R23 = L2^\#$ |
| $R31 = L3^*$                     | $R32 = L2^\#$     | $R33 = L3^*$  |

strength received within the range of 6 meters is reliable and constant. The average value of signal strength which was sent by three different tags and was measured in different ranges is illustrated in Figure 7. As shown in Figure 7, we can establish appropriate signal strength matrix based on the average signal strength received within different ranges.

Theoretically, RFID locating system has the following rule. The more intensive the distribution of readers is, the higher the degree of accuracy is. However, while enjoying higher degree of accuracy, the cost of installing RFID system increases and the problem of signal collision arises. Thus, to consider the factor of accuracy and costs, we conducted several experiments to determine the most cost effective method with the higher degree of accuracy. Table 4 shows the

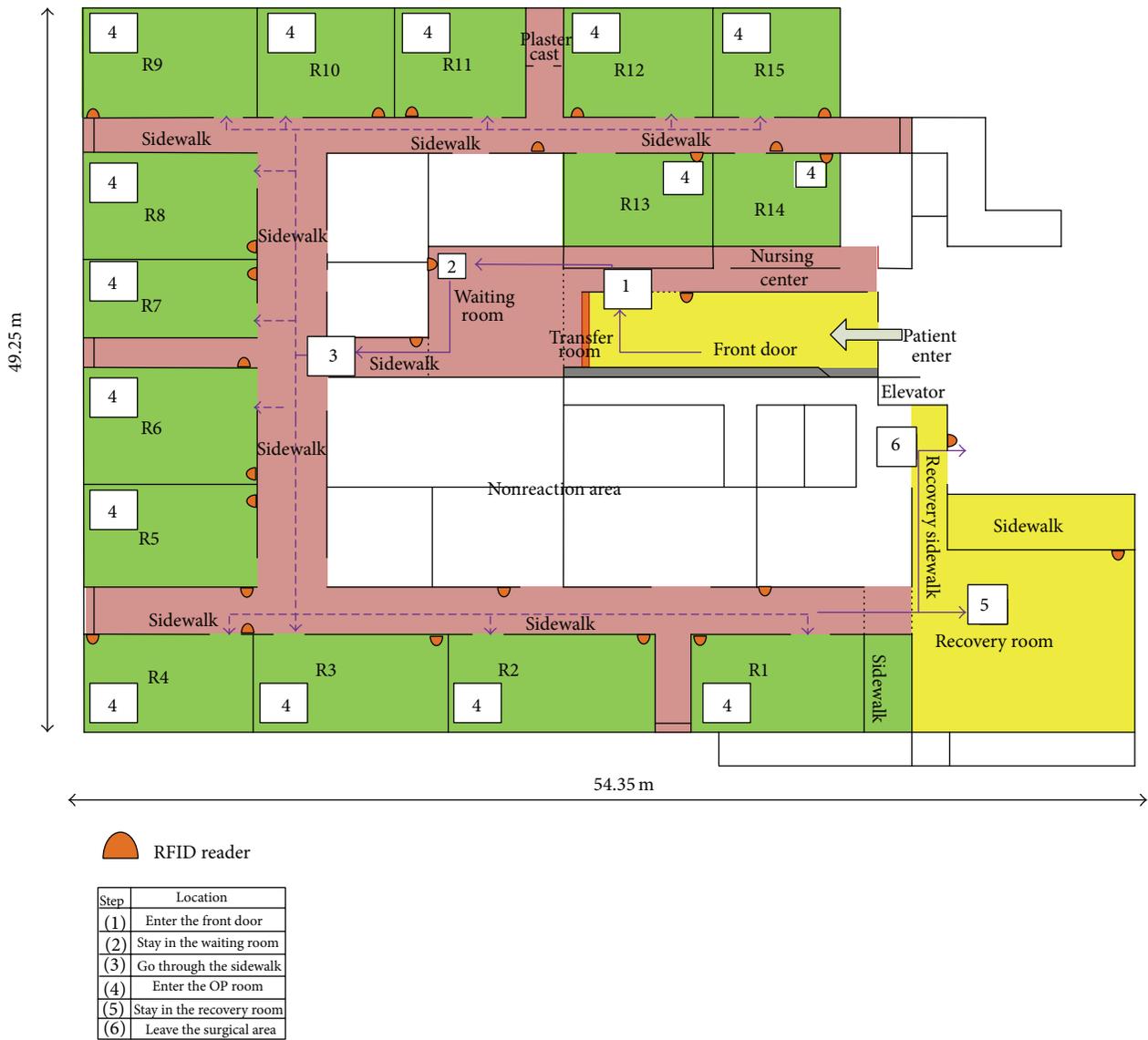


FIGURE 4: The layout of the surgery room of a teaching hospital located in Taipei, Taiwan.

signal strength matrix under different distances between two readers.

The degree of accuracy of the experiment conducted under different distance between two readers ( $D$ ) is listed in Table 5. The degree of accuracy is between 80% and 100% if  $D$  equals 2 and the average value of the degree of accuracy is 94%. The degree of accuracy is between 90% and 100% if  $D$  equals 4. The average value of the degree of accuracy is 98% and it is the highest value among all tests. The degree of accuracy is between 85% and 100% if  $D$  equals 6 and the average value of the degree of accuracy is 97%. The degree of accuracy is between 85% and 100% if  $D$  equals 8 and the average value of the degree of accuracy is 93%. The degree of accuracy decreases dramatically when the distance between two readers is 12 meters and the average value of the degree of accuracy is only 69%. Based on the finding of the experiment, the degree of accuracy can be above 90% if  $D$  is less than

8 meters (including 8 meters). Since the degree of accuracy drops to 69%, we do not consider 12 meters as an option.

Therefore, we simulate the installation of RFID reader/repeater in the surgery room with different distances between two readers. The amount of RFID readers used in setting up the RFID environment under different distance between two readers is shown in Table 6. Since 12 meters is not an option, we only consider the situation where  $D$  is less than 8 meters. According to this table, the more intensive the distribution of RFID readers is, the more the amount of RFID readers is used. When the distance between two readers is 2 meters, number of readers used is almost doubled than the number of readers used when the same distance is 6 meters. Also, when the distance between two readers is 2 meters, the degree of accuracy is less the degree of accuracy when the same distance is 6 meters. Thus, when the distance between two readers is 2 meters, it is not a satisfactory choice.

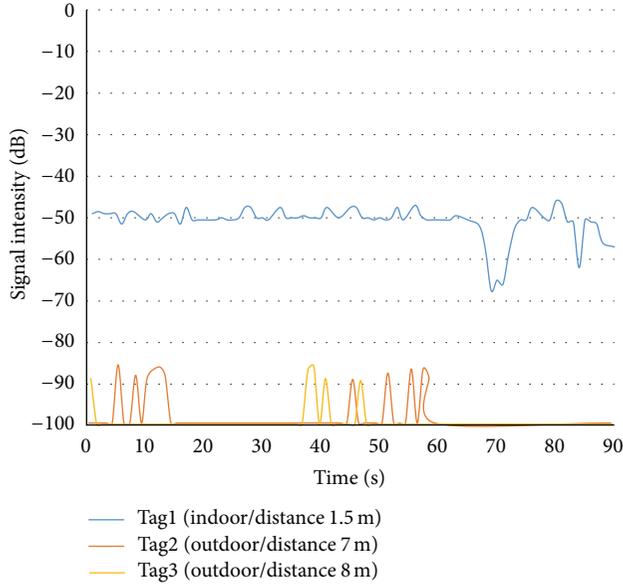


FIGURE 5: Signal Strength received by the reader placed inside the lab room.

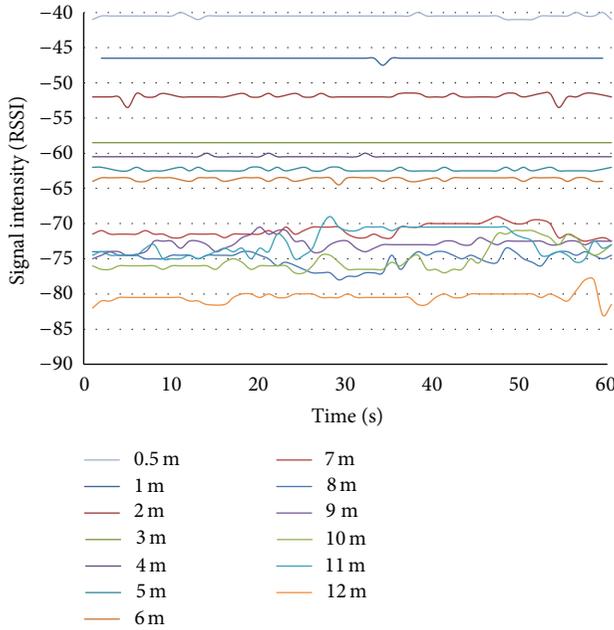


FIGURE 6: Signal strength measured within different ranges.

Furthermore, comparing the cases of 4 meters and 6 meters, the difference between their degrees of accuracy is only one percent while the number of readers used in 6-meters case is less by 8 readers than the 8-meters case. When the issue of costs is considered in setting up the environment, it seems that 6 meters and 8 meters are both acceptable.

A simulated installation of RFID readers with 6-meters distance between two readers in the surgery room is illustrated in Figure 8. In the physical layout, we configured readers, respectively in the front room, the waiting area,

TABLE 4: The signal strength matrix under different distances between two readers.

| (a) Distance between two readers is 2 meters  |                                 |               |
|-----------------------------------------------|---------------------------------|---------------|
| Signal level                                  | Distance between tag and reader | RSS           |
| $0^\dagger$                                   | <0.5 m                          | > -38.5       |
| $L1^\Delta$                                   | 0.5 m~1 m                       | -39 ~ -47.5   |
| $L2^\#$                                       | 1 m~2 m                         | -48 ~ -52.5   |
| $L3^*$                                        | 2 m~3 m                         | -53 ~ -65     |
| (b) Distance between two readers is 4 meters  |                                 |               |
| Signal level                                  | Distance between tag and reader | RSS           |
| $L0^\ddagger$                                 | <1 m                            | > -46         |
| $L1^\Delta$                                   | 1 m~2 m                         | -46.5 ~ -53.5 |
| $L2^\#$                                       | 2 m~4 m                         | -54 ~ -59.5   |
| $L3^*$                                        | 4 m~6 m                         | -60 ~ -67     |
| (c) Distance between two readers is 6 meters  |                                 |               |
| Signal level                                  | Distance between tag and reader | RSS           |
| $L0^\ddagger$                                 | <1 m                            | > -46         |
| $L1^\Delta$                                   | 1 m~3 m                         | -46.5 ~ -58   |
| $L2^\#$                                       | 3 m~6 m                         | -58.5 ~ -64   |
| $L3^*$                                        | 6 m~9 m                         | -64 ~ -77     |
| (d) Distance between two readers is 8 meters  |                                 |               |
| Signal level                                  | Distance between tag and reader | RSS           |
| $L0^\ddagger$                                 | <1 m                            | > -46         |
| $L1^\Delta$                                   | 1 m~4 m                         | -46.5 ~ -59.5 |
| $L2^\#$                                       | 4 m~8 m                         | -60 ~ -69.5   |
| $L3^*$                                        | 8 m~12 m                        | -70 ~ -84     |
| (e) Distance between two readers is 12 meters |                                 |               |
| Signal level                                  | Distance between tag and reader | RSS           |
| $L0^\ddagger$                                 | <1 m                            | > -46         |
| $L1^\Delta$                                   | 1 m~6 m                         | -46.5 ~ -67.5 |
| $L2^\#$                                       | 6 m~12 m                        | -68 ~ -73.5   |
| $L3^*$                                        | 12 m~18 m                       | -74 ~ -90     |

the recovery room, and the operating rooms. The operating rooms are segmented by walls and metal doors. Signals in the operating rooms do not interfere with others. On the pathway, in order to avoid the signal exceeding the receiving range and resulting in the loss of signals, readers are placed on the wall along the pathway. Therefore, as shown in Figure 8, along the pathway, there are at least 1 to 2 readers being able to read the tag signal.

## 5. Conclusion

This paper presents a system that can identify the location of a surgical patient in an operation room based on the different phases in the process of a surgery. In this system, RFID technology that uses active RFID tags and places readers in the environment of an operation room is adopted to monitor and locate the position of surgical patients prior to, in the middle, and after a surgery. The proposed signal

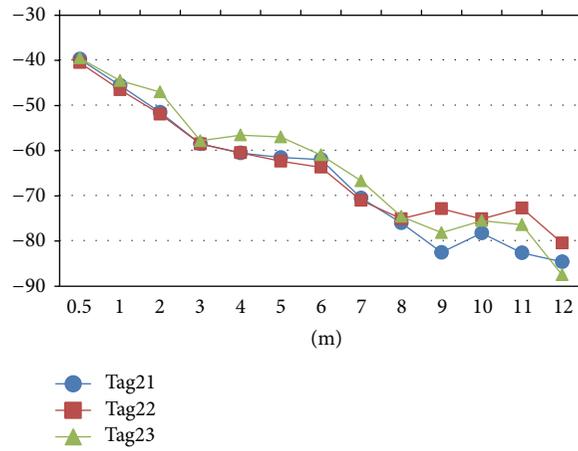


FIGURE 7: Average value of signal strength sent by three tags and measured within different ranges.



FIGURE 8: A simulated installation of RFID readers with 6-meters distance between two readers in the surgery room.

TABLE 5: The degree of accuracy of detecting the location of tags in divided areas under different distances between two readers.

| Distance between two readers | Divided areas |      |      |       |       |        | Average    |
|------------------------------|---------------|------|------|-------|-------|--------|------------|
|                              | A22           | A22U | A22D | A22UL | A22DR | A22DRX |            |
| 2 meters                     | 100%          | 100% | 80%  | 85%   | 100%  | 100%   | <b>94%</b> |
| 4 meters                     | 100%          | 100% | 100% | 90%   | 100%  | 100%   | <b>98%</b> |
| 6 meters                     | 100%          | 95%  | 100% | 100%  | 100%  | 85%    | <b>97%</b> |
| 8 meters                     | 100%          | 95%  | 90%  | 85%   | 85%   | 100%   | <b>93%</b> |
| 12 meters                    | 100%          | 85%  | 100% | 25%   | 25%   | 80%    | <b>69%</b> |

TABLE 6: Number of tags used under different distance between two readers.

| Distance between two readers | Number of tags used           |                                |                              |
|------------------------------|-------------------------------|--------------------------------|------------------------------|
|                              | Number of readers used indoor | Number of readers used outdoor | Total number of readers used |
| 2 meters                     |                               | 44                             | 64                           |
| 4 meters                     | 20                            | 22                             | 42                           |
| 6 meters                     |                               | 14                             | 34                           |
| 8 meters                     |                               | 11                             | 31                           |

level matrix is to solve the problem of sensing overlapping area based on the level of signal strength sent by tags. While moving a hospital bed, there are two kinds of characters: signal strength is in reverse proportion to distance and the movement of a hospital bed is a continuous function shown on a coordinate. These two characters are used to find the location of a hospital bed. A simulation is conducted to imitate the actual situation that occurred in a surgery room. The experiment result shows that this system can accurately identify surgical patient's identification and the position of this patient. By constantly updating patients' location, all relevant units receive correct and newest message about a patient's current status and location to coordinate the work of taking care of a surgical patient smoothly. On one hand, this system can avoid irreparable negligence caused by human error or delayed notification. On the other hand, health care personnel can concentrate more on performing the surgery to provide a higher quality of medical care.

## Conflict of Interests

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

## Acknowledgments

This research is supported by National Science Council of Taiwan, under research Project NSC102-2221-E-227-001. Also, special thanks are due to Mr. Shen-Yuan Tang, the director of the department of Information Management in Shin Kong Wu Ho-Su Memorial Hospital, for his consultation and suggestion on this project.

## References

- [1] L. M. Ni, D. Zhang, and M. R. Souryal, "RFID-based localization and tracking technologies," *IEEE Wireless Communications*, vol. 18, no. 2, pp. 45–51, 2011.
- [2] A. Grover and H. Berghel, "A survey of RFID deployment and security issues," *Journal of Information Processing Systems*, vol. 7, no. 4, pp. 561–580, 2011.
- [3] D. M. Konidala, D. Kim, C. Y. Yeun, and B. Lee, "Security framework for RFID-based applications in smart home environment," *Journal of Information Processing Systems*, vol. 7, no. 1, pp. 111–120, 2011.
- [4] J. Ahn and R. Han, "An indoor augmented-reality evacuation system for the Smartphone using personalized Pedometry," *Human-Centric Computing and Information Sciences*, vol. 2, pp. 1–23, 2012.
- [5] J. Chen, M. B. Salim, and M. Matsumoto, "A single mobile target tracking in voronoi-based clustered wireless sensor network," *Journal of Information Processing Systems*, vol. 7, pp. 17–28, 2011.
- [6] Y. Luo, O. Hoeber, and Y. Chen, "Enhancing Wi-Fi fingerprinting for indoor positioning using human-centric collaborative feedback," *Human-Centric Computing and Information Sciences*, vol. 3, article 2, 2013.
- [7] S. S. Saad and Z. S. Nakad, "A standalone RFID indoor positioning system using passive tags," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1961–1970, 2011.
- [8] R. Want, "An introduction to RFID technology," *IEEE Pervasive Computing*, vol. 5, no. 1, pp. 25–33, 2006.
- [9] J. Hightower, R. Want, and G. Borriello, *SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength*, 2000.
- [10] M. N. Lionel, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: indoor location sensing using active RFID," *Wireless Networks*, vol. 10, no. 6, pp. 701–710, 2004.
- [11] Y. Zhao, Y. Liu, and L. M. Ni, "VIRE: active RFID-based localization using virtual reference elimination," in *Proceedings of the 36th International Conference on Parallel Processing (ICPP '07)*, p. 56, Xi'an, China, September 2007.
- [12] P. Krishnan, A. S. Krishnakumar, W. Ju, C. Mallows, and S. Ganu, "A system for LEASE: location estimation assisted by stationary emitters for indoor RF wireless networks," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 1001–1011, Hong Kong, March 2004.
- [13] C.-S. Wang, X.-M. Huang, and M.-Y. Hung, "Adaptive rfid positioning system using signal level matrix," in *Proceedings of*

*the International Conference on Sensor Networks, Information, and Ubiquitous Computing*, Singapore, 2010.

- [14] R. Hosaka, "Feasibility study of convenient automatic identification system of medical articles using LF-band RFID in hospital," *Systems and Computers in Japan*, vol. 35, no. 10, pp. 74–82, 2004.
- [15] D.-H. Shih, H.-S. Chiang, B. Lin, and S.-B. Lin, "An embedded mobile ECG reasoning system for elderly patients," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 3, pp. 854–865, 2010.
- [16] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems*, vol. 10, no. 1, pp. 91–102, 1992.
- [17] C. A. Thompson, "Radio frequency tags for identifying legitimate drug products discussed by tech industry," *American Journal of Health-System Pharmacy*, vol. 61, no. 14, pp. 1430–1432, 2004.
- [18] J. A. Fisher and T. Monahan, "Tracking the social dimensions of RFID systems in hospitals," *International Journal of Medical Informatics*, vol. 77, no. 3, pp. 176–183, 2008.
- [19] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.
- [20] I. D'Souza, W. Ma, and C. Notobartolo, "Real-time location systems for hospital emergency response," *IT Professional*, vol. 13, no. 2, pp. 37–43, 2011.
- [21] G. B. Gentili, F. Dori, and E. Iadanza, "Dual-frequency active RFID solution for tracking patients in a children's hospital. Design method, test procedure, risk analysis, and technical solution," *Proceedings of the IEEE*, vol. 98, no. 9, pp. 1656–1662, 2010.
- [22] J. K.-Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, pp. 15–20, 2012.
- [23] D. Kang, K. Kang, H. Lee, E. Ko, and J. Lee, "A systematic design tool of context aware system for ubiquitous healthcare service in a smart home," in *Proceedings of the International Conference on Future Generation Communication and Networking (FGCN '07)*, pp. 49–54, Jeju, Republic of Korea, December 2007.
- [24] J. E. Bardram and N. Nørskov, "A context-aware patient safety system for the operating room," in *Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp '08)*, pp. 272–281, September 2008.
- [25] R. Tesoriero, J. A. Gallud, M. Lozano, and V. M. R. Penichet, "Using active and passive RFID technology to support indoor location-aware systems," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, pp. 578–583, 2008.

## Research Article

# Taxonomy and Evaluations of Low-Power Listening Protocols for Machine-to-Machine Networks

**Kwang-il Hwang and Sung-Hyun Yoon**

*Department of Embedded Systems Engineering, Incheon National University, Incheon 402-772, Republic of Korea*

Correspondence should be addressed to Kwang-il Hwang; [hkwangil@incheon.ac.kr](mailto:hkwangil@incheon.ac.kr)

Received 2 April 2014; Accepted 4 June 2014; Published 8 July 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 K.-i. Hwang and S.-H. Yoon. 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.

Even though a lot of research has made significant contributions to advances in sensor networks, sensor network protocols, which have different characteristics according to the target application, might confuse machine-to-machine (M2M) network designers when they choose the protocol most suitable for their specific applications. Therefore, this paper provides a well-defined taxonomy of low-power listening protocols by examining in detail the existing low-power sensor network protocols and evaluation results. It will also be very useful for helping M2M designers understand specific features of low-power media access control protocols as they design new M2M networks.

## 1. Introduction

Machine-to-machine (M2M) networks enable creation of the Internet of Things, which interconnects via the Internet physical things equipped with various sensors and actuators. Mitsui et al. [1] presented various M2M applications based on sensor network technologies. A typical M2M architecture is basically composed of three domains: the server, the Internet, and the sensors. In particular, the sensor domain is the most important, aggregating data from physical sensors and accessing the Internet via 3G or 4G M2M gateways. Like a sensor network, an M2M sensor domain requires well-structured and energy-efficient network protocols among distributed sensors using short range communications. Much research has already been conducted on sensor network protocols [2], making significant contributions towards advances in automated sensor networks [3–5]. However, having too many sensor network protocols causes confusion for M2M designers as they choose the protocol most suitable for their specific applications. Furthermore, most of the literature on sensor network protocols is too theoretical, requires a lot of specific assumptions, and is not easy to apply to practical M2M sensor domains.

Sensor media access control (MAC) protocols can be categorized into random-based, slot (schedule-) based, time

division multiple access- (TDMA-) based, random/TDMA hybrids, and low-power listening (LPL) methods. In particular, LPL-based MAC protocols can be considered the most suitable type for M2M sensor domains, because they provide a low duty cycle and low implementation complexity. Therefore, there has been substantial research on LPL protocols. Each one shows different characteristics and operations, as described in Table 1. Therefore, this paper aims to provide a well-defined taxonomy of low-power listening protocols by examining in detail the existing low-power sensor network protocols, introducing an M2M communication model and then evaluating performance with respect to data aggregation time and energy consumption in terms of an M2M communication model.

The remainder of this paper is organized as follows. A taxonomy of LPL protocols is presented in Section 2. Section 3 analyzes each LPL protocol in terms of an M2M communications model. Section 4 summarizes numerical results and Section 5 provides concluding remarks.

## 2. A Taxonomy of Low-Power Listening Protocols

*2.1. Trigger Source (Preamble versus Packet).* The main idea of LPL is to asynchronously trigger a receiver that is alternating

TABLE 1: LPL MAC protocols.

| Protocol     | Features                                                                                                                                                                                                                                  |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| B-MAC [6]    | (i) Berkeley MAC<br>(ii) LPL with check time, back-off window size, and power-down policy in the application level<br>(iii) Advanced clear channel assessment (CCA) for dealing with random noise                                         |
| Wise-MAC [7] | (i) Improved LPL, remembering neighbors' polling schedules<br>(ii) Sends short preamble when the receiver wakes up                                                                                                                        |
| X-MAC [8]    | (i) Upgraded B-MAC protocol<br>(ii) Divides long preamble into two parts (micropreamble/receiver address) to solve overhearing                                                                                                            |
| SpeckMAC [9] | (i) Consists of SpeckMAC-B, SpeckMAC-D<br>(ii) Consecutive data frame, wake-up packet<br>(iii) Sender accesses receiver with 3 bytes preamble in the packet frame                                                                         |
| RI-MAC [10]  | (i) Receiver-initiated MAC protocol<br>(ii) Receiver sends periodic beacon frame and sender sends data frame if beacon frame is received                                                                                                  |
| BoX-MAC [11] | (i) Cross-layer MAC protocol using PHY, link layer<br>(ii) Consists of two parts (BoX-MAC-1/BoX-MAC-2)<br>(iii) Goes into sleep state in back-off time<br>(iv) Less wake-up time than X-MAC                                               |
| MX-MAC [12]  | (i) An LPL variant of CSMA-MPS<br>(ii) Compatible with X-MAC and SpeckMAC<br>(iii) Consecutive packet transmission instead of short preamble strobe in X-MAC<br>(iv) Sends ACK when packet is received to solve X-MAC's early ACK problem |
| A-MAC [13]   | (i) Receiver-initiated MAC protocol<br>(ii) Using hardware-generated acknowledgment (HACK) for more efficient energy consumption<br>(iii) Saves neighbors' LPL schedules<br>(iv) Deals with hidden terminal problem                       |

between wake-up and sleep states to detect a wake-up signal from a sender. Therefore, receivers can save much more energy by removing idle listening periods. Some protocols, such as B-MAC, WISE-MAC, and X-MAC, use a preamble as a trigger source. On the other hand, SpeckMAC, RI-MAC, BoX-MAC, MX-MAC, and A-MAC trigger receivers by transmitting a consecutive packet. More specifically, RI-MAC, MX-MAC, A-MAC, and SPEC-MAC-D utilize a data packet for the trigger, and SpeckMAC-B, BoX-MAC-1, and BoX-MAC-2 utilize short wake-up packets before data transmission.

*2.2. Initiation Method (Receiver-Initiated versus Source-Initiated).* LPL protocols can also be categorized into source-initiated and receiver-initiated methods, according to which one begins the transmission request. RI-MAC and A-MAC are receiver-initiated protocols but the rest of the protocols are source-initiated protocols.

*2.3. Adaptivity (Adaptive versus Deterministic).* B-MAC, SpeckMAC, RI-MAC, A-MAC, and BoX-MAC-1 always transmit triggering signals for predetermined fixed duration, but some protocols, such as WISE-MAC, X-MAC, MX-MAC, and BoX-MAC-2, transmit variable triggering signals depending on when a receiver is triggered.

*2.4. Schedule (Schedule versus Nonschedule).* To reduce data pending time more, some protocols, such as WISE-MAC and MX-MAC, use schedule-based triggering by exchanging wake-up time information among neighbors.

### 3. M2M Communication Model

In this section, an M2M communication model is presented, and then each LPL protocol is analyzed in terms of the M2M model.

*3.1. System Model.* Generally, M2M is composed of a concentrator, which is a centralized device to connect the M2M sensor domain to the Internet, and M2M devices, which are equipped with various sensors or actuators. In an M2M sensor domain, devices form either a star or a peer-to-peer topology for multihop communications. Data from each device are aggregated in the concentrator and transmitted to a corresponding server via the Internet. To consider a practical M2M system, each protocol and algorithm should be able to execute their tasks with off-the-shelf radio frequency (RF) modems (TI CC430, CC2420, RadioPulse MG2400, etc.) and MCUs.

*3.2. Data Model.* The most popular data models for M2M are the *periodic report model* and the *request-oriented model*. In the periodic report model, each device transmits data to a concentrator periodically, and the model is generally used for unidirectional data aggregation. By contrast, the request-oriented model allows bidirectional communication between the concentrator and devices. In the data model, a server (user) can request a concentrator to aggregate real-time sensor data in the sensor domain. The concentrator also triggers and transmits server requests to the devices. Each device replies to the concentrator, and the responses from

TABLE 2: Notations.

| Notation                | Description                                   | Value                                                                                                                                                              |
|-------------------------|-----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $T_{PS}$                | Preamble sensing time                         | 15.6 milliseconds                                                                                                                                                  |
| $T_{WS}$                | Wake-up sensing time                          | 183 milliseconds                                                                                                                                                   |
| $T_{FS}$                | Frame sensing time                            | 183 milliseconds                                                                                                                                                   |
| $T_{BS}$                | Beacon sending time                           | 90 milliseconds                                                                                                                                                    |
| $T_S$                   | Sleep duration                                | Variable milliseconds                                                                                                                                              |
| $T_P$                   | Preamble transmission time for device trigger | (i) Long preamble (B-MAC)<br>(ii) Short preamble (X-MAC)<br>(iii) Variable (Wise-MAC)                                                                              |
| $T_{WP}$                | Wake-up packet time for trigger               | 90 milliseconds                                                                                                                                                    |
| $T_{DP}$                | Data packet time for trigger                  | 150 milliseconds                                                                                                                                                   |
| $T_{BL}$                | Beacon listen time                            | Random                                                                                                                                                             |
| $T_B$                   | Back-off period                               | 1 milliseconds                                                                                                                                                     |
| $T_L$                   | Listen period                                 | Variable                                                                                                                                                           |
| $T_{DT}$                | Data transmission time                        | 200 milliseconds                                                                                                                                                   |
| $T_{ACK}$               | ACK transmission time                         | 90 milliseconds                                                                                                                                                    |
| $T_{DPT}$               | Data pending time                             | Random                                                                                                                                                             |
| $T_{DC}$                | Duty-cycle time                               | $= T_{PS} + T_S$ (preamble-based)<br>$= T_{WS} + T_S$ (wake-up packet-based)<br>$= T_{FS} + T_S$ (packet-based)<br>$= \text{beacon interval}$ (receiver-initiated) |
| $T_{\text{tot-active}}$ | Total active duration                         | $T_{A-PPS} + T_{A-aggre}$ (i)                                                                                                                                      |
| $T_{\text{tot-sleep}}$  | Total sleep duration                          | $T_{\text{interval}} - T_{\text{tot-active}}$ (ii)                                                                                                                 |
| $T_{\text{interval}}$   | Request interval                              | 1 hour (=3600 seconds = 3600000 milliseconds)                                                                                                                      |
| $I_A$                   | Current consumption in active state           | 0.0061944 mA                                                                                                                                                       |
| $I_S$                   | Current consumption in active state           | 0.0000083 mA                                                                                                                                                       |
| $T_{A-aggre}$           |                                               | Active duration during data aggregation time                                                                                                                       |
| $T_{A-PPS}$             |                                               | Active duration in a duty cycle                                                                                                                                    |
| $T_{AW}$                | Acknowledgement waiting period                | 100 milliseconds                                                                                                                                                   |
| $V$                     | Supply voltage                                | 3.5 V                                                                                                                                                              |

devices are aggregated in the concentrator and transmitted to the server.

**3.3. Energy Model.** For M2M networks, energy conservation is one of the most critical challenges, as it is in sensor networks. It is important to note that in order to save energy, each device should remain active only for required duration, and the rest of the time should go to sleep. Therefore, when calculating the energy consumption of each device, we need to know the total active duration,  $T_{\text{tot-active}}$  (i) and the total sleep duration,  $T_{\text{tot-sleep}}$  (ii) in a request interval,  $T_{\text{interval}}$ . By using (i) and (ii), the energy consumption for each device can be expressed as follows:

## 4. Numerical Analysis

Now, we numerically analyze each LPL protocol in terms of M2M communication models. In particular, we focus on data aggregation time, which is the total time required to aggregate data from all devices with respect to a request. Table 2 presents notations used for our numerical analysis.

**4.1. B-MAC.** B-MAC is a representative LPL protocol utilizing a preamble for the receiver trigger. As shown in Figure 1, each device repeats a short time wake-up for  $T_{PS}$  to detect the preamble transmission and then sleeps for  $T_S$  per  $T_{DC}$ . A sender that wants to send data first transmits a long preamble for  $T_P$  to trigger the receiver that is performing periodic preamble sensing (PPS) before data transmission. Each preamble transmission can be detected by all devices within communication range of a sender, and all nodes that detect the preamble transmission, as well as the intended receiver, have to remain active for  $T_L$ , until the preamble transmission ends.

**4.1.1. Periodic Report.** Since each device should send its data to the concentrator on the predetermined schedule, the report time of each device is as follows:

$$T_{\text{resp}} = T_B + T_P + T_{DT} + T_{ACK}. \quad (1)$$

Therefore, the total report time of  $n$  nodes is

$$T_{\text{B-MAC}(P)} = n * T_{\text{resp}} = n * (T_B + T_P + T_{DT} + T_{ACK}). \quad (2)$$



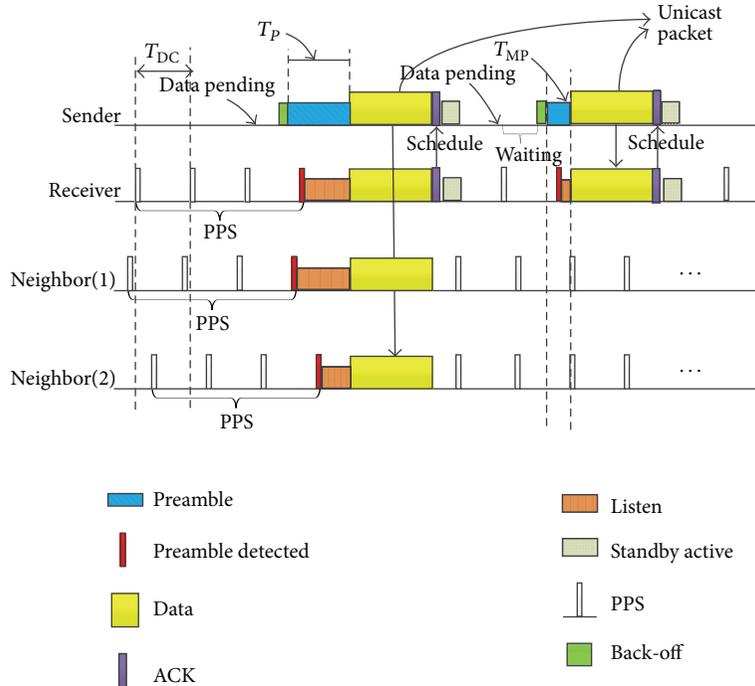


FIGURE 2: WISE-MAC.

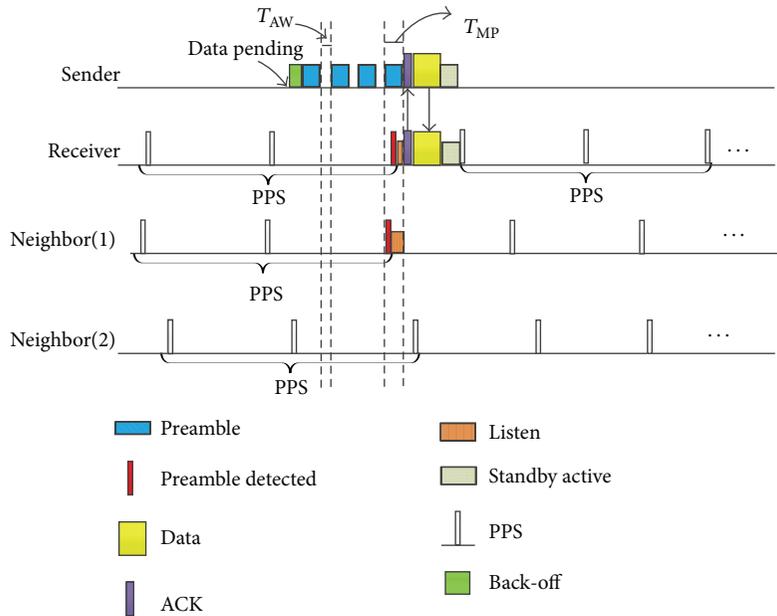


FIGURE 3: X-MAC.

4.3.2. *Request-Oriented.* The required time for a concentrator to transmit its request to devices is

$$T_{req} = T_B + m * (T_{MP} + T_{AW}) + T_{DT} + T_{ACK}. \quad (11)$$

And, unlike B-MAC and WISE-MAC, X-MAC cannot trigger all devices with a single preamble transmission. Therefore,

the request of the concentrator must be transmitted as many times as the number of devices. So the total aggregation time of  $n$  nodes per request is

$$\begin{aligned} T_{X-MAC(R)} &= n * (T_{req} + T_{resp}) = 2 * n * T_{req} \\ &= 2 * n * (T_B + m * (T_{MP} + T_{AW}) + T_{DT} + T_{ACK}). \end{aligned} \quad (12)$$

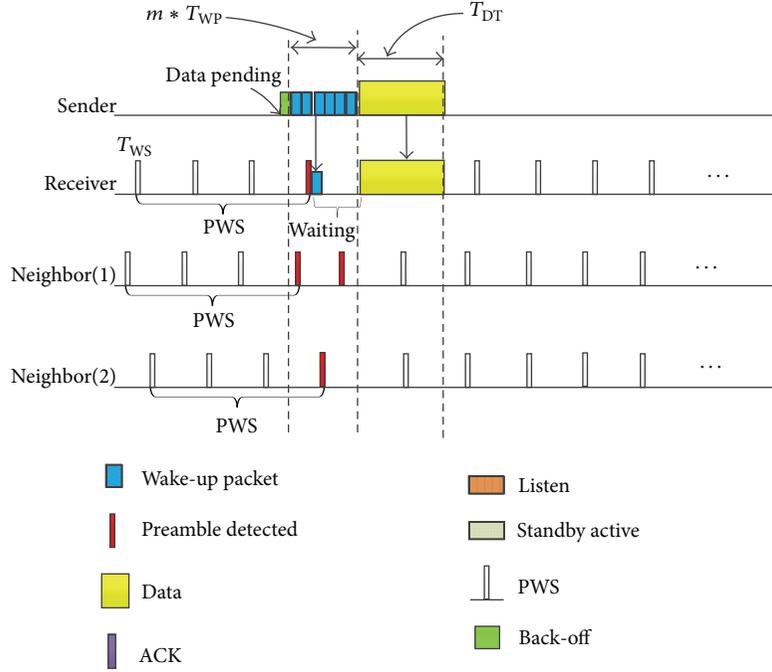


FIGURE 4: SpeckMAC-B.

4.4. *SpeckMAC-B*. Instead of a preamble transmission, SpeckMAC-B transmits consecutive wake-up packets to trigger devices performing periodic wake-up-signal sensing (PWS), as shown in Figure 4. A wake-up packet contains a destination ID and a time stamp, which represents data packet transmission time information. Therefore, a device that listens to a wake-up packet during PWS goes to sleep until the beginning of data transmission wakes up and then receives data from the sender. Devices that listen to a wake-up packet but that are not the intended receiver go to sleep and continue to perform PWS.

4.4.1. *Periodic Report*. The report time of a SpeckMAC-B device is as follows:

$$T_{\text{resp}} = T_B + m * T_{\text{WP}} + T_{\text{DT}}, \quad (13)$$

where  $m$  is a maximum time to trigger the receiver and  $m \leq T_{\text{DC}}$ .

Therefore, the total report time of  $n$  nodes is

$$T_{\text{SPECK-MAC-B}(P)} = n * T_{\text{resp}} = n * (T_B + m * T_{\text{WP}} + T_{\text{DT}}). \quad (14)$$

4.4.2. *Request-Oriented*. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * T_{\text{WP}} + T_{\text{DT}}. \quad (15)$$

And, like B-MAC or WISE-MAC, a single request packet can trigger all devices, so the total aggregation time of  $n$  nodes per request is

$$\begin{aligned} T_{\text{SPECK-MAC-B}(R)} &= T_{\text{req}} + n * T_{\text{resp}} = (n + 1) * T_{\text{req}} \\ &= (n + 1) * (T_B + m * T_{\text{WP}} + T_{\text{DT}}). \end{aligned} \quad (16)$$

4.5. *SpeckMAC-D*. Instead of the wake-up packet transmission used in SpeckMAC-B, SpeckMAC-D enables fast data reception by utilizing consecutive data packets. Each SpeckMAC-D device performs periodic frame sensing (PFS) for  $T_{\text{FS}}$  to receive a data frame, as shown in Figure 5.

4.5.1. *Periodic Report*. The report time of a SpeckMAC-D device is as follows:

$$T_{\text{resp}} = T_B + T_{\text{DP}} * m, \quad (17)$$

where  $m$  is a maximum time to trigger the receiver and  $m \leq T_{\text{DC}}$ .

Therefore, the total report time of  $n$  nodes is

$$T_{\text{SPECK-MAC-D}(P)} = n * T_{\text{resp}} = n * (T_B + T_{\text{DP}} * m). \quad (18)$$

4.5.2. *Request-Oriented*. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + T_{\text{DP}} * m. \quad (19)$$

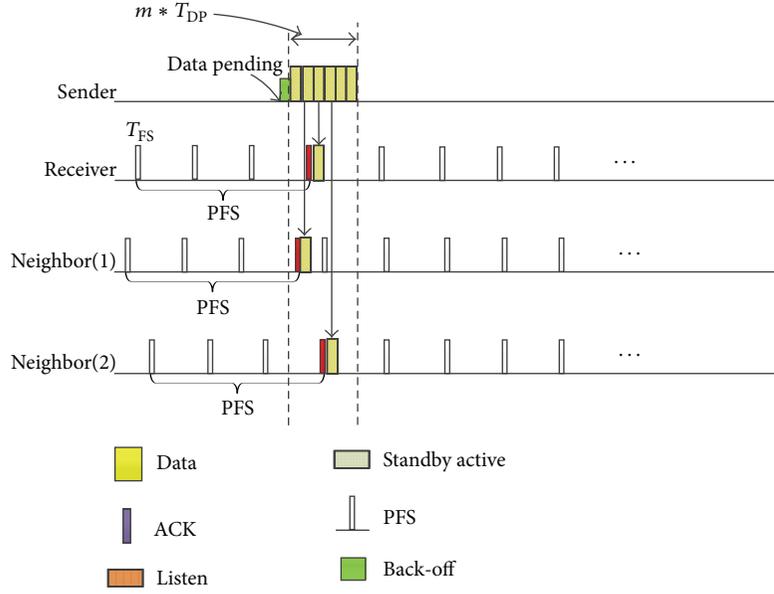


FIGURE 5: SpeckMAC-D.

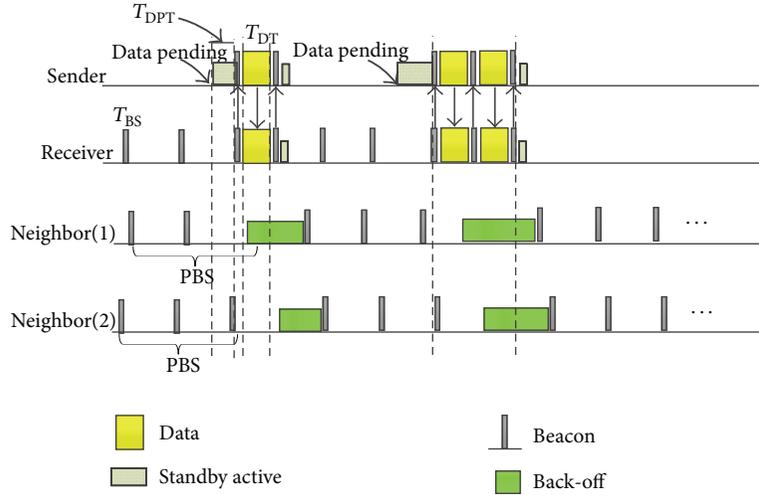


FIGURE 6: RI-MAC.

And  $T_{req}$  is equal to  $T_{resp}$ . So the total aggregation time of  $n$  nodes per request is

$$\begin{aligned}
 T_{SPECK-MAC-D(R)} &= T_{req} + n * T_{resp} \\
 &= (n + 1) * T_{req} = (n + 1) * (T_B + T_{DP} * m).
 \end{aligned}
 \tag{20}$$

**4.6. RI-MAC.** RI-MAC is a representative receiver-initiated LPL protocol. Each RI-MAC device basically performs periodic beacon sending (PBS). A sender first switches to reception (RX) mode and waits to receive the beacon of a corresponding receiver. As soon as a corresponding beacon is received, the sender transmits data and then goes back to PBS. The receiver of the data acknowledges a beacon, as shown in Figure 6.

**4.6.1. Periodic Report.** The report time of an RI-MAC device is as follows:

$$T_{resp} = T_{DT} + 2 * T_{BS}.
 \tag{21}$$

Therefore, the total report time of  $n$  nodes is

$$T_{RI-MAC(P)} = n * T_{resp} = n * (T_{DT} + 2 * T_{BS}).
 \tag{22}$$

**4.6.2. Request-Oriented.** The required time for a concentrator to transmit its request to devices is

$$T_{req} = T_{DT} + 2 * T_{BS}.
 \tag{23}$$

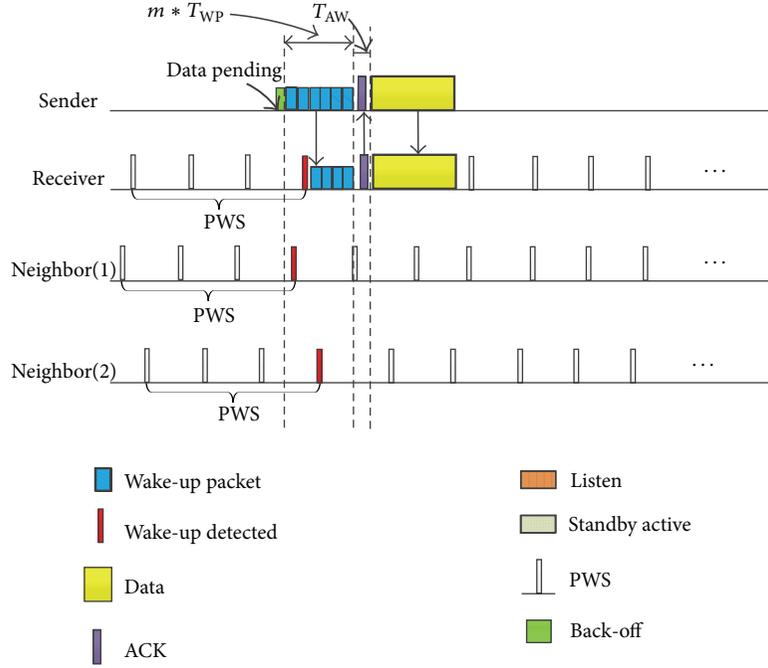


FIGURE 7: BoX-MAC-1.

And  $T_{\text{req}}$  is equal to  $T_{\text{resp}}$ . So the total aggregation time of  $n$  nodes per request is

$$\begin{aligned} T_{\text{RI-MAC}(R)} &= n * (T_{\text{req}} + T_{\text{resp}}) = 2 * n * T_{\text{req}} \\ &= 2 * n * (T_{\text{DT}} + 2 * T_{\text{BS}}). \end{aligned} \quad (24)$$

**4.7. BoX-MAC-1.** As shown in Figure 7, BoX-MAC-1 is one of the packet-based LPL protocols, and  $T_{\text{AW}}$  (to wait for ACK of the receiver) is followed by consecutive wake-up packets. Then, on reception of the ACK, the data are transmitted.

**4.7.1. Periodic Report.** The report time of a BoX-MAC-1 device is as follows:

$$T_{\text{resp}} = T_B + m * T_{\text{WP}} + T_{\text{ACK}} + T_{\text{DT}}, \quad (25)$$

where  $m$  is a maximum time to trigger the receiver and  $m \leq T_{\text{DC}}$ .

Therefore, the total report time of  $n$  nodes is

$$T_{\text{BOX-MAC-1}(P)} = n * T_{\text{resp}} = n * (T_B + m * T_{\text{WP}} + T_{\text{ACK}} + T_{\text{DT}}). \quad (26)$$

**4.7.2. Request-Oriented.** The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * T_{\text{WP}} + T_{\text{ACK}} + T_{\text{DT}}. \quad (27)$$

And since data transmission is started only when an ACK is received, the request of the concentrator must be transmitted

as many times as the number of devices. So the total aggregation time of  $n$  nodes per request is as follows:

$$\begin{aligned} T_{\text{BOX-MAC-1}(R)} &= n * (T_{\text{req}} + T_{\text{resp}}) \\ &= 2 * n * T_{\text{req}} \\ &= 2 * n * (T_B + m * T_{\text{WP}} + T_{\text{ACK}} + T_{\text{DT}}). \end{aligned} \quad (28)$$

**4.8. BoX-MAC-2.** BoX-MAC-2 is one of the wake-up, packet-based LPL protocols. However, unlike BoX-MAC-1 or SpeckMAC-B utilizing consecutive wake-up-packet transmissions, a sender waits for ACK from the receiver for  $T_{\text{AW}}$ , per wake-up transmission, as shown in Figure 8. Therefore, a sender repeats wake-up packet transmission and RX for ACK until receiving the ACK.

**4.8.1. Periodic Report.** The report time of a BoX-MAC-2 device is as follows:

$$T_{\text{resp}} = T_B + m * (T_{\text{WP}} + T_{\text{AW}}) + T_{\text{DT}} + 2 * T_{\text{ACK}}, \quad (29)$$

where  $m$  is a maximum time to trigger the receiver and  $m \leq T_{\text{DC}}$ .

Therefore, the total report time of  $n$  nodes is

$$\begin{aligned} T_{\text{BOX-MAC-2}(P)} &= n * T_{\text{resp}} \\ &= n * (T_B + m * (T_{\text{WP}} + T_{\text{AW}}) + T_{\text{DT}} + 2 * T_{\text{ACK}}). \end{aligned} \quad (30)$$



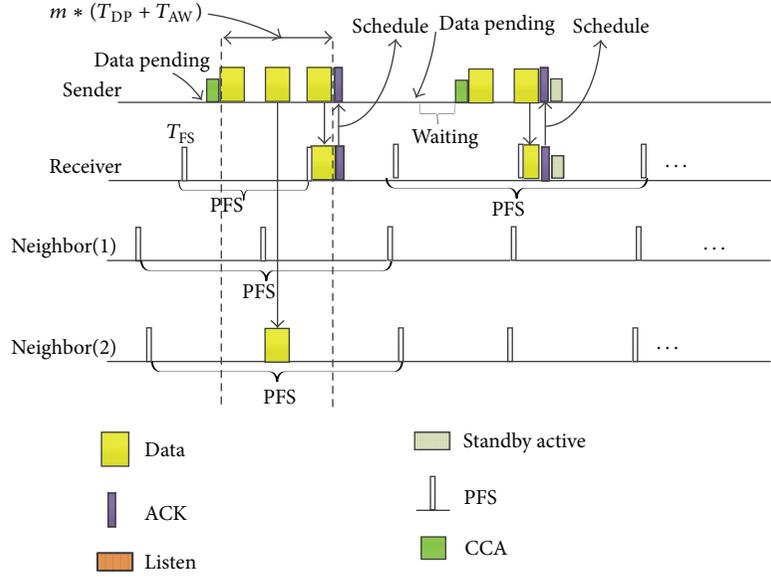


FIGURE 9: MX-MAC.

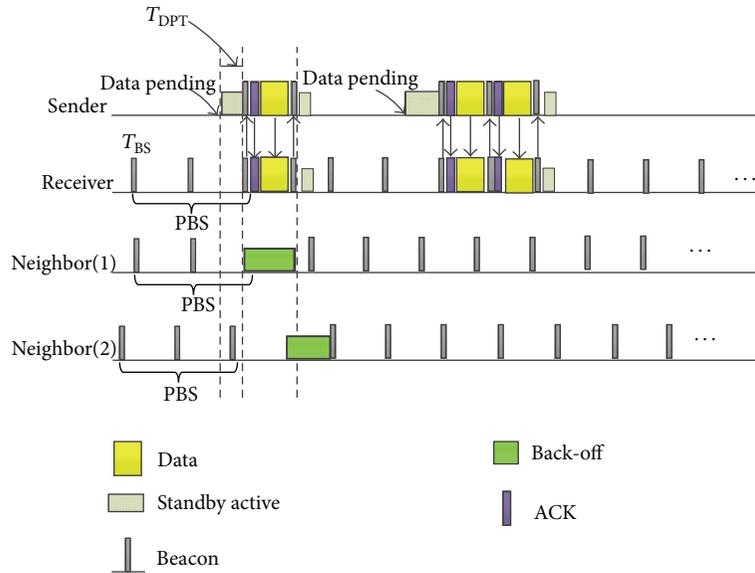


FIGURE 10: A-MAC.

4.10.2. *Request-Oriented.* The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}. \quad (39)$$

$T_{\text{req}}$  is equal to  $T_{\text{resp}}$ , and each request is paired with each response. So the total aggregation time of  $n$  nodes per request is

$$\begin{aligned} T_{\text{A-MAC}(R)} &= n * (T_{\text{req}} + T_{\text{resp}}) \\ &= 2 * n * T_{\text{req}} \\ &= 2 * n * (T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}). \end{aligned} \quad (40)$$

## 5. Summary and Concluding Remarks

Based on the taxonomy presented in Section 2 and the numerical analysis presented in Section 4, Table 3 presents a summary of the taxonomy and evaluation results regarding data aggregation time and energy consumption in terms of M2M communication models. In addition, protocol complexity is evaluated in terms of time synchronization requirements, memory usage, and ability to be implemented with an off-the-shelf RF modem and MCU.

First, in terms of data aggregation time, without regarding wake-up source type, while adaptive and schedule-based protocols such as WISE-MAC and MX-MAC show fast data aggregation time, preamble-based or receiver-initiated

TABLE 3: Summary of taxonomy and evaluations.

| Protocol     | Wake-up  |        | Initiation |        | Adaptivity |               | Schedule  |             | Data aggregation time |                  | Energy consumption |        | Protocol complexity |         |
|--------------|----------|--------|------------|--------|------------|---------------|-----------|-------------|-----------------------|------------------|--------------------|--------|---------------------|---------|
|              | Preamble | Packet | Receiver   | Source | Adaptive   | Deterministic | Scheduled | Nonschedule | Periodic report       | Request-oriented | Concentrator       | Device |                     | Network |
| B-MAC        | ✓        |        |            | ✓      | ✓          |               |           | ✓           | ○                     | ○                | ●                  | ●      | ●                   | ○       |
| WISE-MAC     | ✓        |        |            | ✓      |            |               | ✓         |             | ●                     |                  | ●                  | ●      | ●                   | ●       |
| X-MAC        | ✓        |        |            | ✓      | ✓          |               | ✓         |             |                       |                  | ●                  | ●      | ●                   | ●       |
| SPECK-MAC(B) |          | ✓      |            | ✓      | ✓          |               | ✓         | ✓           | ○                     |                  |                    |        |                     | ○       |
| SPECK-MAC(D) |          | ✓      |            | ✓      | ✓          |               | ✓         | ✓           |                       |                  |                    |        |                     | ○       |
| RI-MAC       |          | ✓      |            | ✓      | ✓          |               | ✓         | ✓           |                       |                  |                    |        |                     | ○       |
| BOX-MAC(1)   |          | ✓      | ✓          | ✓      | ✓          |               | ✓         | ✓           | ○                     |                  |                    |        |                     | ○       |
| BOX-MAC(2)   |          | ✓      |            | ✓      | ✓          |               | ✓         | ✓           | ○                     |                  |                    | ○      | ○                   | ○       |
| MX-MAC       |          | ✓      |            | ✓      | ✓          |               | ✓         | ✓           |                       |                  |                    | ○      | ○                   | ○       |
| A-MAC        |          | ✓      | ✓          | ✓      | ✓          |               | ✓         | ✓           | ○                     | ●                |                    |        |                     | ○       |

●: A : B : C : O: D.

protocols like B-MAC, RI-MAC, and A-MAC present long aggregation times. This is because adaptive LPL protocols are capable of coping with a receiver's reaction through feedback during wake-up duration, compared with deterministic protocols utilizing fixed-size wake-up duration without regard to the receiver's reaction.

In terms of energy efficiency, while preamble-based protocols, such as B-MAC, WISE-MAC, and X-MAC present superior energy efficiency, data packet-based LPL protocols like BoX-MAC and MX-MAC present high energy consumption. Since preamble detection duration is considerably shorter than the data reception duration, the preamble-based protocols can operate with a very short duty cycle.

In terms of protocol complexity, deterministic or receiver-initiated protocols have relatively low complexity, whereas adaptive and schedule-based protocols, such as WISE-MAC, SpeckMAC-B, and MX-MAC, have high complexity because they require tight time synchronization and management for neighbors' PPS times. In addition, X-MAC (which transmits a short preamble in which ID information is contained) is not possible to implement with an off-the-shelf RF modem.

Lastly, we expect the summarized taxonomy will provide a useful guideline for understanding the specific features of LPL protocols and for designing a new M2M network.

## 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 Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education, Science and Technology (2012R1A1A2041271).

## References

- [1] H. Mitsui, H. Kambe, and H. Koizumi, "Student experiments for learning basic M2M technologies by implementing sensor network systems," in *Proceedings of the 9th International Conference on Information Technology Based Higher Education and Training (ITHET '10)*, pp. 268–275, Cappadocia, Turkey, April–May 2010.
- [2] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The evolution of MAC protocols in wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 101–120, 2013.
- [3] J. Liu and S. H. Chung, "An efficient load balancing scheme for multi-gateways in wireless mesh networks," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 365–378, 2013.
- [4] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, no. 13, pp. 1–17, 2013.
- [5] M. Yoon, Y. K. Kim, and J. W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, 2013.
- [6] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, New York, NY, USA, November 2004.
- [7] A. El-Hoiydi and J. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Proceedings of the 9th International Symposium on Computers and Communications (ISCC '04)*, pp. 244–251, June–July 2004.
- [8] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 307–320, New York, NY, USA, November 2006.
- [9] K.-J. Wong and D. Arvind, "SpeckMAC: low-power decentralized MAC protocols for low data rate transmissions in specknets," in *Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality (REALMAN '06)*, pp. 71–78, New York, NY, USA, May 2006.
- [10] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (SenSys '08)*, pp. 1–14, New York, NY, USA, November 2008.
- [11] D. Moss and P. Levis, "BoX-MAC: exploiting physical and link layer boundaries in low-power networking," Tech. Rep. SING-08-00, 2008.
- [12] C. J. Merlin and W. B. Heinzelman, "Schedule adaptation of low-power-listening protocols for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 672–685, 2010.
- [13] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. M. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," *ACM Transactions on Sensor Networks*, vol. 8, no. 3, 2012.

## Research Article

# Task Balanced Workflow Scheduling Technique considering Task Processing Rate in Spot Market

Daeyong Jung,<sup>1</sup> JongBeom Lim,<sup>1</sup> JoonMin Gil,<sup>2</sup> Eunyoung Lee,<sup>3</sup> and Heonchang Yu<sup>1</sup>

<sup>1</sup> Department of Computer Science Education, Korea University, 321A, Lyceum, Anam-Dong, Seongbuk-Gu, Seoul 136-701, Republic of Korea

<sup>2</sup> School of Information Technology Engineering, Catholic University of Daegu, Daegu, Republic of Korea

<sup>3</sup> Department of Computer Science, Dongduk Women's University, Seoul, Republic of Korea

Correspondence should be addressed to Heonchang Yu; yuhc@korea.ac.kr

Received 21 January 2014; Accepted 4 June 2014; Published 29 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Daeyong Jung 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.

Recently, the cloud computing is a computing paradigm that constitutes an advanced computing environment that evolved from the distributed computing. And the cloud computing provides acquired computing resources in a pay-as-you-go manner. For example, Amazon EC2 offers the Infrastructure-as-a-Service (IaaS) instances in three different ways with different price, reliability, and various performances of instances. Our study is based on the environment using spot instances. Spot instances can significantly decrease costs compared to reserved and on-demand instances. However, spot instances give a more unreliable environment than other instances. In this paper, we propose the workflow scheduling scheme that reduces the out-of-bid situation. Consequently, the total task completion time is decreased. The simulation results reveal that, compared to various instance types, our scheme achieves performance improvements in terms of an average combined metric of 12.76% over workflow scheme without considering the processing rate. However, the cost in our scheme is higher than an instance with low performance and is lower than an instance with high performance.

## 1. Introduction

In recent years, due to the increased interests in cloud computing, many cloud projects and commercial systems such as Amazon EC2 [1] have been implemented. Cloud computing provides many benefits including easy access to user data, ease of management for users, and the reduction of costs. And cloud computing services provide a high level of scalability of computing resources combined with internet technology to many customers [2, 3]. In most cloud services, the concept of an instance unit is used to provide users with resources in a cost-efficient manner. There are many different cloud computing providers and each offers different layers of services. This paper focuses on Infrastructure-as-a-Service (IaaS) platforms that allow clients access to massive computational resources in the form of instances [4–7].

Generally, cloud computing resources use reliable on-demand instances. On-demand instances allow the user to pay for computing capacity by hour, with no long-term

commitments. This frees users from the costs and complexities of planning, purchasing, and maintaining hardware and transforms what are usually large fixed costs into much smaller variable costs [1]. However, on-demand instance may incur upper cost than other instances such as reserved instance and spot instance. We focus on spot instances in unreliable environment. For such a reason, if you have time flexibility for executing applications, spot instances can significantly decrease your Amazon EC2 costs [8, 9]. For task completion, therefore, spot instances may incur lower cost than on-demand instances.

The spot instance is configured by spot market-based cloud environment. In the spot instance environment, variations of spot prices are dependent on the supply and demand of spot instances. The environment affects the successful completion or failure of tasks depending on the variation of spot prices. Spot prices have a market structure and follow the law of demand and supply. Therefore, cloud services (Amazon EC2) can provide a spot instance when a user's bid is

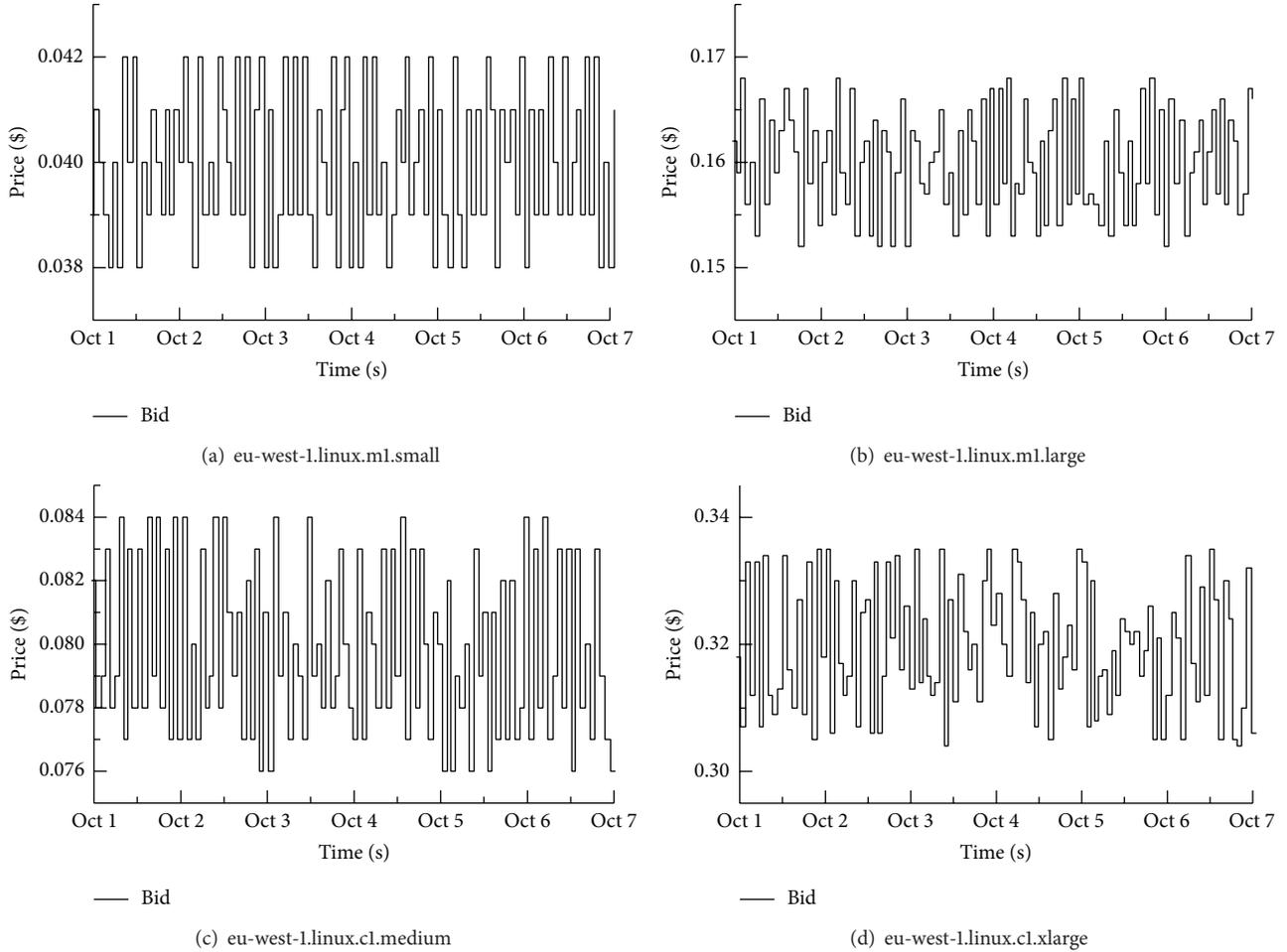


FIGURE 1: Price history of EC2's spot instances.

higher than the current spot price. Further, a running instance stops when a user's bid becomes less than or equal to the current spot price. After a running instance stops, it restarts when a user's bid becomes greater than the current spot price [10–12].

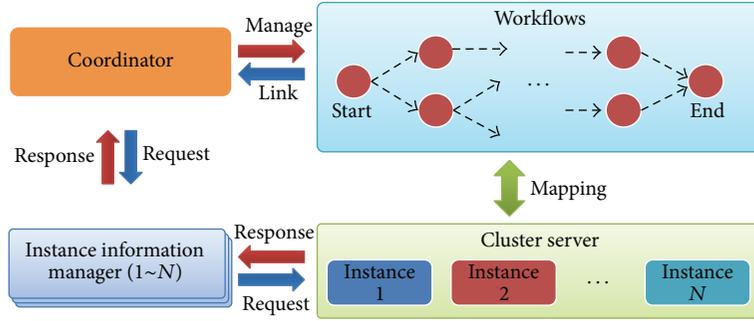
In particular, the scientific application makes the current common of workflow. However, the spot instance-based cloud computing takes various performances. In spot instance, an available execution time depends on a spot price. The spot price changes periodically based on user's demand and supply. The completion time for the same amount of a task varies according to the performance of an instance. In particular, the failure time of each instance differs according to the user's bid and the performance in an instance. Therefore, we solve the problem that a completion time of a task in an instance increases when a failure occurs. For an efficient execution of a task, the task is divided into subtasks on various types of available instances. We analyze information of the task and the instance from price history. We estimate the size of task and the information of an available instance from the analyzed data. We create workflow using each available instance and the size of a task. As a consequence, we propose

the scheduling scheme using workflow to solve job execution problem and considering task processing rate. And we execute user's job at the boundary of selected instances and expand the suggested user budget.

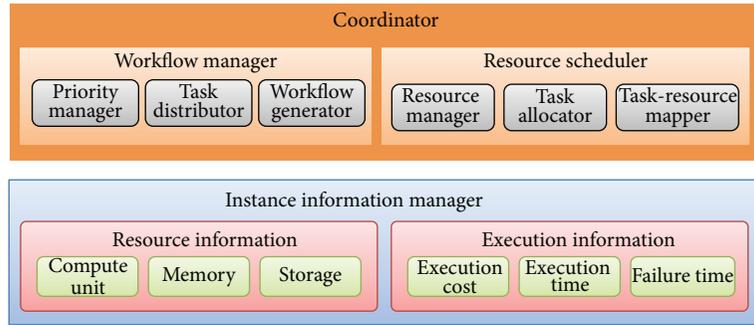
## 2. Background and Related Works

In this section, we begin by describing the workflow model focusing on spot instances. Firstly, we explain the background of spot instances in cloud environments. In the spot instances environment [8, 9], there are numerous studies on fault tolerance [10–12] and workflow scheduling [13, 14].

*2.1. Spot Instances.* Amazon EC2 offers the IaaS instances in three different ways with different price, reliability, and various performances of instances. Those are reserved instances, on-demand instances, and spot instances. In case of reserved instances, a user pays a yearly fee and receives a discount on hourly rates. And, in case of on-demand instances, a user pays the fee on hourly rate. In spot instances, a user determines the user's bid and spot price decides spot market based on the user's demand and supply. Our scheduling focuses on offering



(a) The mapping relation of workflow and instances



(b) The constitution of coordinator and manager

FIGURE 2: Workflow environment.

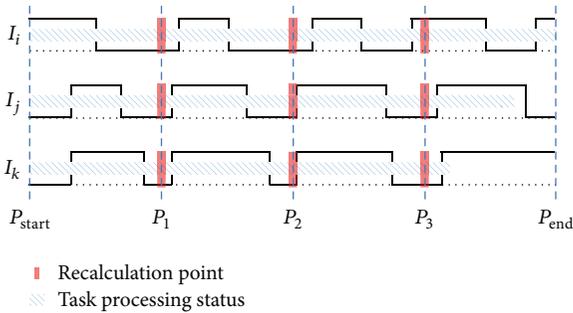


FIGURE 3: The recalculating point of the task size.

services at the boundary of spot instances. Spot instances give an unreliable environment compared to reserved and on-demand instances. However, spot instances can significantly decrease user's costs compared to other instances. The spot price in spot instance is based on market structure and law of demand and supply. Therefore, cloud service can provide a spot instance when a user's bid is higher than the current spot price. If the user's bid exceeds the current market price, the user runs the instance. However, if the market price exceeds the user's bid, the instance is terminated and the partial hours are not charged. And the spot system immediately stops the spot instance without any notice to the user. After a running instance stops, the instance restarts when a user's bid is greater than the current spot price. An example of spot history is shown in Figure 1. This figure shows examples

of fluctuations of spot price for standard instance (m1-small and m1-large) and high-CPU instance (c1-medium and c1-large) during 7 days in October 2010 [15].

**2.2. Fault Tolerance.** On the fault tolerance side, two similar studies (hour-boundary checkpointing [10] and rising edge-driven checkpointing [11]) proposed enforcing fault tolerance in cloud computing with spot instance. Based on the actual price history of EC2 spot instances, they compared several adaptive checkpointing schemes in terms of monetary costs and job execution time. In hour-boundary checkpointing, the checkpointing operation is performed in the hour boundary, and a user pays the bidding price on an hourly basis. In rising edge-driven checkpointing, checkpointing operation is performed when the price of the spot instance is raised and the price is less than the user's bid. However, two schemes have problems that the costs and task completion time are increased due to increase of the number of checkpoints. To solve these problems, in our previous study [12], we proposed the checkpointing scheme using checkpoint thresholds based on rising-driven checkpointing. The checkpointing is basically performed using two thresholds, price and time, based on the expected execution time according to the price history. Therefore, we propose a workflow system to apply the previous proposed checkpointing.

**2.3. Workflow Scheduling.** A workflow is a model that represents complex problems with structures such as directed acyclic graphs (DAG). Workflow scheduling is a kind of

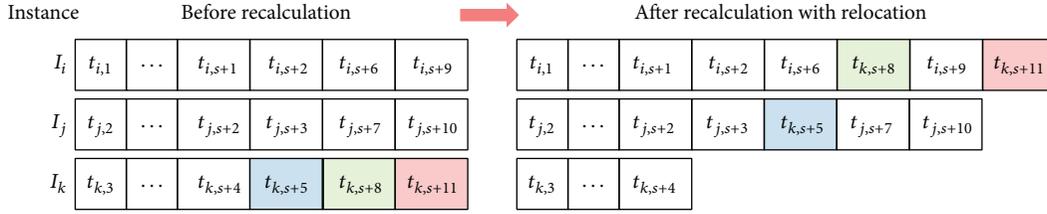


FIGURE 4: The recalculation operation of the assigned task.

global task scheduling as it focuses on mapping and managing the execution of interdependent tasks on shared resources. However, the existing workflow scheduling methods have the limited scalability and are based on centralized scheduling algorithm. Consequently, these methods are not suitable for spot instance-based cloud computing. In spot instance, the job execution has to consider available time and cost of an instance. Fully decentralized workflow scheduling system determines the instance to use the chemistry-inspired model in community cloud platform [13]. A throughput maximization strategy is designed for transaction-intensive workflow scheduling that does not support multiple workflows [14]. Our proposed scheduling guarantees an equal task distribution to available instances in spot instance-based cloud computing. And the scheduling method performs redistribution of the tasks based on task processing rate.

### 3. Proposed Workflow System

**3.1. System Architecture.** Our proposed scheme is expanded from our previous work [12] and includes a workflow scheduling algorithm. Figure 2(a) presents the relation of workflows and instances and Figure 2(b) shows the constitution of coordinator and manager. Figure 2 illustrates the roles of the instance information manager, the workflow manager, and the resource scheduler. The instance information manager obtains information for the job allocation and resource management. The information includes VM specifications in each instance and the execution-related information such as the execution costs, execution completion time, and failure time. The execution-related information is calculated by using the selected VM based on spot history. The workflow manager and resource scheduler extract the needed execution-related information from the instance information manager. First, the workflow manager generates the workflow for the requested job. The generated workflow determines the task size according to the VM performance, the execution time and costs, and the failure time when the selected instance is used. Secondly, the resource scheduler manages the resource and allocates the task to handle the job. Resource and task managements are needed in order to reallocate tasks when the resource cannot get the information for the task and when the task has a fault during execution.

**3.2. Workflow Scheduling Technique considering Task Processing Rate.** The scheduling scheme is depicted in Figure 3.

The instances  $I_i$ ,  $I_j$ , and  $I_k$  mean high, medium, and low performance, respectively. The instance  $I_k$  belongs to a positive group and the other two instances ( $I_i$ ,  $I_j$ ) belong to a negative group. The scheduler distributes a task size to allocate available instances and considers performance of instances. Task size recalculation points divide the fourth quarter based on the expected task execution time and recalculate each quarter except for the last quarter. The task size rate is determined based on the average of task execution time of each instance within the recalculated point. And the modified task size in each instance is allocated to consider the task size rate.

Figure 4 shows the recalculation point of the task size from the  $P_1$  position in Figure 3. In Figure 4, we assume that the processing rate of instances is proportional to the performance of instances. The left side of Figure 4, “before recalculation,” represents the tasks assigned to each instance. The right side, “after recalculation with relocation,” shows the result of task migration based on the average task execution time in each instance. After a recalculation operation, we perform the rearrangement of tasks. The rearrangement method sorts tasks in increasing order of their indices.

To design the above model, our proposed scheme uses the workflow in spot instance and its purpose is to minimize job processing time within the suggested cost of user. The task size is determined by considering the availability and performance of each instance in order to minimize the job processing time. The available time is estimated by the execution time and cost using the price history of spot instances to improve the performance and stability of task processing. The estimated data is determined to assign the amount of tasks to each instance. Our proposed scheme reduces the out-of-bid situation and improves the job execution time. However, total cost is higher than when not using workflow.

Our task distribution method determines the task size in order to allocate a task to a selected instance. Based on a compute-unit and an available state, the task size of an instance  $I_i(T_i)$  is calculated as follows:

$$T_i = \left( \frac{U_{I_i} \times A_{I_i}}{\sum_{i=1}^N (U_{I_i} \times A_{I_i})} \right) \times \frac{1}{U_{I_i}} \times T_{\text{request}} \times U_{\text{baseline}}, \quad (1)$$

where  $T_{\text{request}}$  represents the total size of tasks required for executing a user request. In an instance  $I_i$ ,  $U_{I_i}$  and  $A_{I_i}$  represent the compute-unit and the available state, respectively. The available state  $A_{I_i}$  can be either 0 (unavailable) or 1 (available).

```

(1) Boolean S_flag = false // a flag representing occurrence of a task execution
(2) while (search user's job) do
(3) if (require job execution by the user) then
(4) take the cost and total execution time by the user;
(5) S_flag = true;
(6) end if
(7) if (S_flag) then
(8) invoke initial_workflow (); // thread function
(9) while (task execution does not finish) do
(10) if (meet the recalculation point by instance) then
(11) invoke recalculation_workflow (); // thread function
(12) end if
(13) end while
(14) end if
(15) end while

```

ALGORITHM 1: Workflow scheduling algorithm.

```

(1) Thread_Function initial_workflow () begin
(2) forall instance $I_i \in \text{Ins}$ do
(3) retrieve an instance information to meet the user's requirement in an
 instance I_i ;
(4) analyze an available execution time and cost in an instance I_i ;
(5) store the analyzed available instance to a queueinstance;
(6) end forall
(7) calculate on priority list for the priority job allocation;
(8) forall instance $I_i \in \text{queue}_{\text{instance}}$ do
(9) allocate tasks to the instance I_i ;
(10) end forall
(11) end Thread_Function
(12) Thread_Function recalculation_workflow () begin
(13) forall instance $I_i \in \text{Ins}$ do
(14) retrieve the information $T_{\text{rate}}^{I_i}$ to an instance I_i ;
(15) calculate the modified task size;
(16) end forall
(17) end Thread_Function

```

ALGORITHM 2: Workflow recalculation function.

We use the instance rate  $T_{\text{rate}}^{I_i}$  for determining the criteria to divide groups.  $T_{\text{rate}}^{I_i}$  represents the unit taken for the processing of a task size in the instance  $I_i$ . Consider

$$T_{\text{rate}}^{I_i} = \frac{T_{\text{execution}}^{I_i} + T_{\text{failure}}^{I_i}}{T_{\text{execution}}^{I_i}}, \quad (2)$$

where  $T_{\text{execution}}^{I_i}$  and  $T_{\text{failure}}^{I_i}$  represent the task execution time and the task failure time, respectively.

And we define the avg to classify groups. The avg is the average of available instances such as  $T_{\text{rate}}^{\text{avg}}$  and  $T_{\text{avg}}$  which represent the average of the  $T_{\text{rate}}^{I_i}$  and  $T_{I_i}$ , respectively. The set of instances is classified into two groups, positive and negative, based on  $T_{\text{rate}}^{\text{avg}}$ . The positive group  $G_P$  is the set of instances with  $T_{\text{rate}}^{I_i}$  greater than  $T_{\text{rate}}^{\text{avg}}$ . Consider

$$G_P = \{I_i T_{\text{rate}}^{I_i} \geq T_{\text{rate}}^{\text{avg}}, 1 \leq i \leq N\}. \quad (3)$$

We calculate the task size to transfer from instance  $I_i$  ( $Tr_{I_i}$ ) in  $G_P$  as follows:

$$Tr_{I_i} = \frac{[(T_{I_i} - T_{\text{execution}}^{I_i}) \times I_{\text{rate}}^{I_i} - (T_{\text{avg}} - T_{\text{execution}}^{\text{avg}}) \times I_{\text{rate}}^{\text{avg}}]}{I_{\text{rate}}^{I_i}}, \quad 1 \leq i \leq N. \quad (4)$$

In group  $G_P$ , the task size of each instance  $I_i$  is given as  $T_{I_i \in G_P}'$ . We are able to get  $T_{I_i \in G_P}'$  by considering  $Tr_{I_i}$  after the transfer operation:

$$T_{I_i \in G_P}' = T_{I_i \in G_P} - Tr_{I_i}, \quad 1 \leq i \leq N. \quad (5)$$

The negative group  $G_N$  is the set of instances  $I_i$  with  $T_{\text{rate}}^{I_i}$  less than  $T_{\text{rate}}^{\text{avg}}$ . Consider

$$G_N = \{I_i T_{\text{rate}}^{I_i} < T_{\text{rate}}^{\text{base}}, 1 \leq i \leq N\}. \quad (6)$$

TABLE 1: Information of resource types.

| Instance type name       | Compute unit | Virtual cores      | Spot price min | Spot price average | Spot price max |
|--------------------------|--------------|--------------------|----------------|--------------------|----------------|
| m1.small (Standard)      | 1 EC2        | 1 core (1 EC2)     | \$0.038        | \$0.040            | \$0.053        |
| m1.large (Standard)      | 4 EC2        | 2 cores (2 EC2)    | \$0.152        | \$0.160            | \$0.168        |
| m1.xlarge (Standard)     | 8 EC2        | 4 cores (2 EC2)    | \$0.076        | \$0.080            | \$0.084        |
| c1.medium (High-CPU)     | 5 EC2        | 2 cores (2.5 EC2)  | \$0.304        | \$0.323            | \$1.52         |
| c1.xlarge (High-CPU)     | 20 EC2       | 8 cores (2.5 EC2)  | \$0.532        | \$0.561            | \$0.588        |
| m2.xlarge (High-Memory)  | 6.5 EC2      | 2 cores (3.25 EC2) | \$0.532        | \$0.561            | \$0.588        |
| m2.2xlarge (High-Memory) | 13 EC2       | 4 cores (3.25 EC2) | \$0.532        | \$0.561            | \$0.588        |
| m2.4xlarge (High-Memory) | 26 EC2       | 8 cores (3.25 EC2) | \$1.064        | \$1.22             | \$1.176        |

TABLE 2: Parameters and values for simulation.

| Simulation parameter | Task time interval | Baseline  | Distribution time | Merge time | Checkpoint time | Recovery time |
|----------------------|--------------------|-----------|-------------------|------------|-----------------|---------------|
| Value                | 43,200 (s)         | m1.xlarge | 300 (s)           | 300 (s)    | 300 (s)         | 300 (s)       |

The tasks are allocated according to the instance performance  $U_{I_i}$ . The task size to receive  $R_{I_i}$  is allocated according to the task size of each instance  $I_i$ . In the group  $G_N$ , the task size of each instance is given as  $T'_{I_i \in G_N}$ . After the receive operation,  $R_{I_i}$  is added to  $T_{I_i \in G_N}$ . Consider

$$R_{I_i} = \frac{U_{I_i}}{\sum_{i \in G_N} U_{I_i}} \times \sum_{i \in G_p} (Tr_{I_i} \times U_{I_i}) \times \frac{1}{U_{I_i}}, \quad (7)$$

$$1 \leq i \leq N,$$

$$T'_{I_i \in G_N} = T_{I_i \in G_N} + R_{I_i}, \quad 1 \leq i \leq N.$$

We propose a workflow scheduling algorithm based on the above equations. Algorithms 1 and 2 show the workflow scheduling algorithm and the workflow recalculation function, respectively.

#### 4. Performance Evaluation

The simulations were conducted using the history data obtained from Amazon EC2 spot instances [15]. The history data before 10-01-2010 was used to extract the expected execution time and failure occurrence probability for our checkpointing scheme. The applicability of our scheme was tested using the history data after 10-01-2010.

In the simulations, one type of spot instance was applied to show the effect of an analysis—task time—on the performance. Table 1 shows various resource types used in Amazon EC2. In this table, resource types comprise a number of different instance types. First, standard instances offer

a basic resource type. Second, high-CPU instances offer more compute-units than other resources and can be used for compute-intensive applications. Finally, high-memory instances offer more memory capacity than other resources and can be used for high-throughput applications, including database and memory caching applications. Under the simulation environments, we compare the performance of our proposed scheme with that of the existing schemes without distributions of tasks in terms of various analyses according to the task time.

Table 1 shows various information of resource type in each instance and Table 2 shows the parameters and values for simulation. The information of spot price is extracted from 11-30-2009 to 01-23-2011 in spot history. The user's bid is taken by the spot price average from information of spot price. The task size is decided by compute-unit rate based on baseline. Initially, the baseline denotes an instance m1.xlarge. For example, the task size of an instance m1.small is calculated by the following:

$$T_{m1.small} = \frac{U_{m1.xlarge}}{U_{m1.small}} \times T_{\text{original task}}. \quad (8)$$

*4.1. Comparison Results of Each Instance before Applying Workflow.* Figure 5 shows the simulation results about each instance. We consider performance condition of each instance. Each instance sets user's bid to take the spot price average in Table 2. Figure 5 presents the execution time and costs according to various instances types. The instance with high performance reduces the execution time but spends higher cost than the instance with low performance.

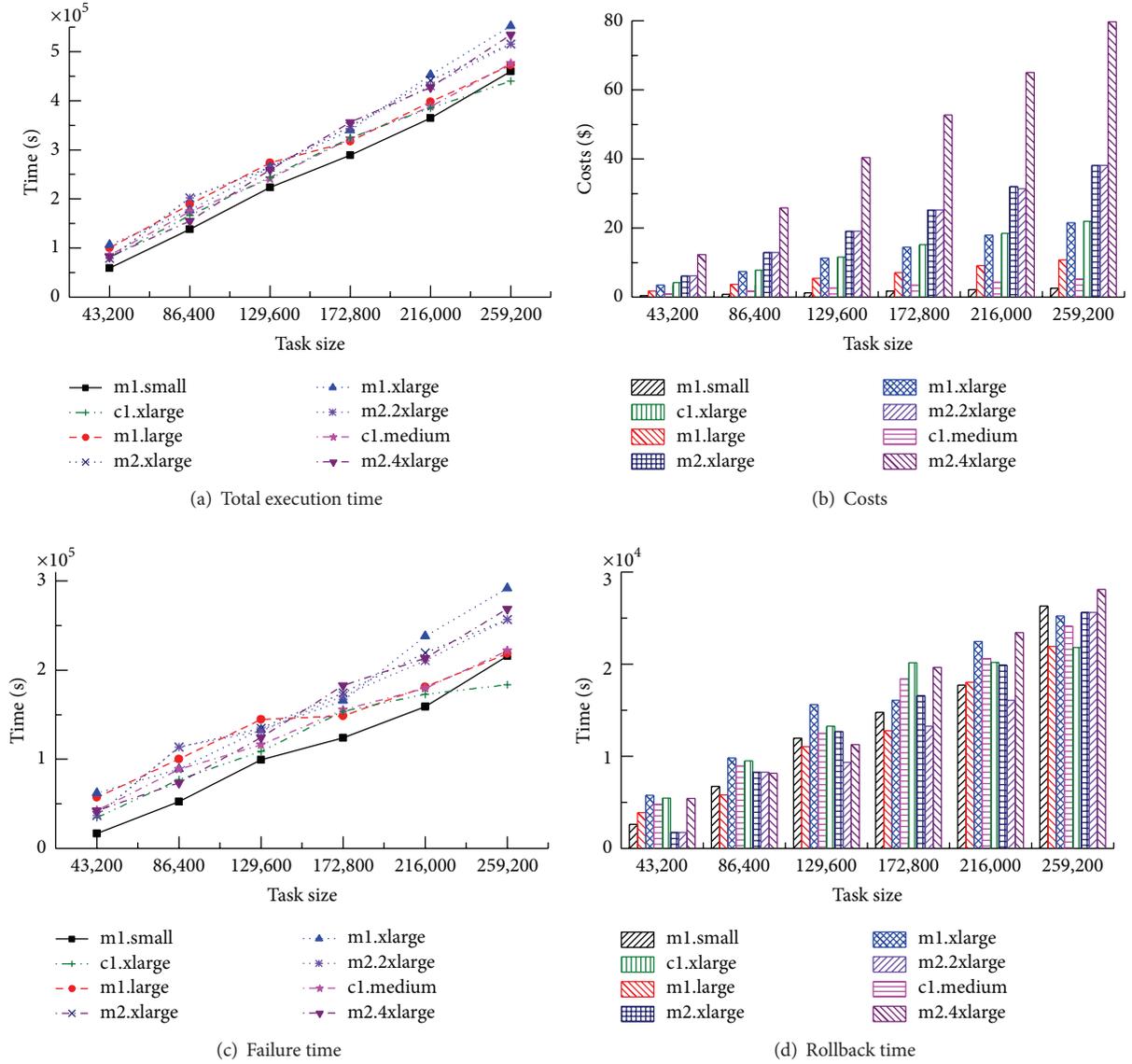


FIGURE 5: Simulation result in each instance.

As, in Figure 5(a), the total execution time increases, Figure 5(c) describes that the failure time increases. Figure 5(d) shows the rollback time in each instance. Rollback time is the time interval between a failure occurrence time and the last checkpoint time.

**4.2. Comparison Results after Applying Workflow.** Figure 6 shows the simulation results about the task distribution. Figure 6(a) shows the total execution time for each instance and Total. In the figures,  $Total_T$  denotes the total time taken for distributing and merging tasks.  $Total_C$  denotes the sum of costs of task execution in each instance. The total execution time of the  $Total_T$  achieves performance improvements in terms of an average execution time of 81.47% over the shortest execution time in each task time interval. In Figure 6(b), the cost in our scheme increases an average of \$11.64 compared

to an instance m1.small and reduces an average of \$32.87 compared to an instance m2.4xlarge. A failure time of Figure 6(c) and a rollback time of Figure 6(d) are smaller than those of Figures 5(c) and 5(d).

Figure 7 shows the execution results of workflow based on the task processing rate after applying our proposed scheme. Figures 6(a) and 7(a) show that the total execution time is reduced by an average of 18.8% after applying our scheme compared to not applying it. Figures 6(c) and 7(c) show that the failure time after applying our proposed scheme was increased by 6.68% compared to before applying it. However, in Figures 6(d) and 7(d), the rollback time after applying our proposed scheme showed an average performance improvement of 4.3% when compared to the rollback time without applying it. The rollback time is calculated from a failure point to the last checkpoint time. Figures 6(b) and 7(b) show that

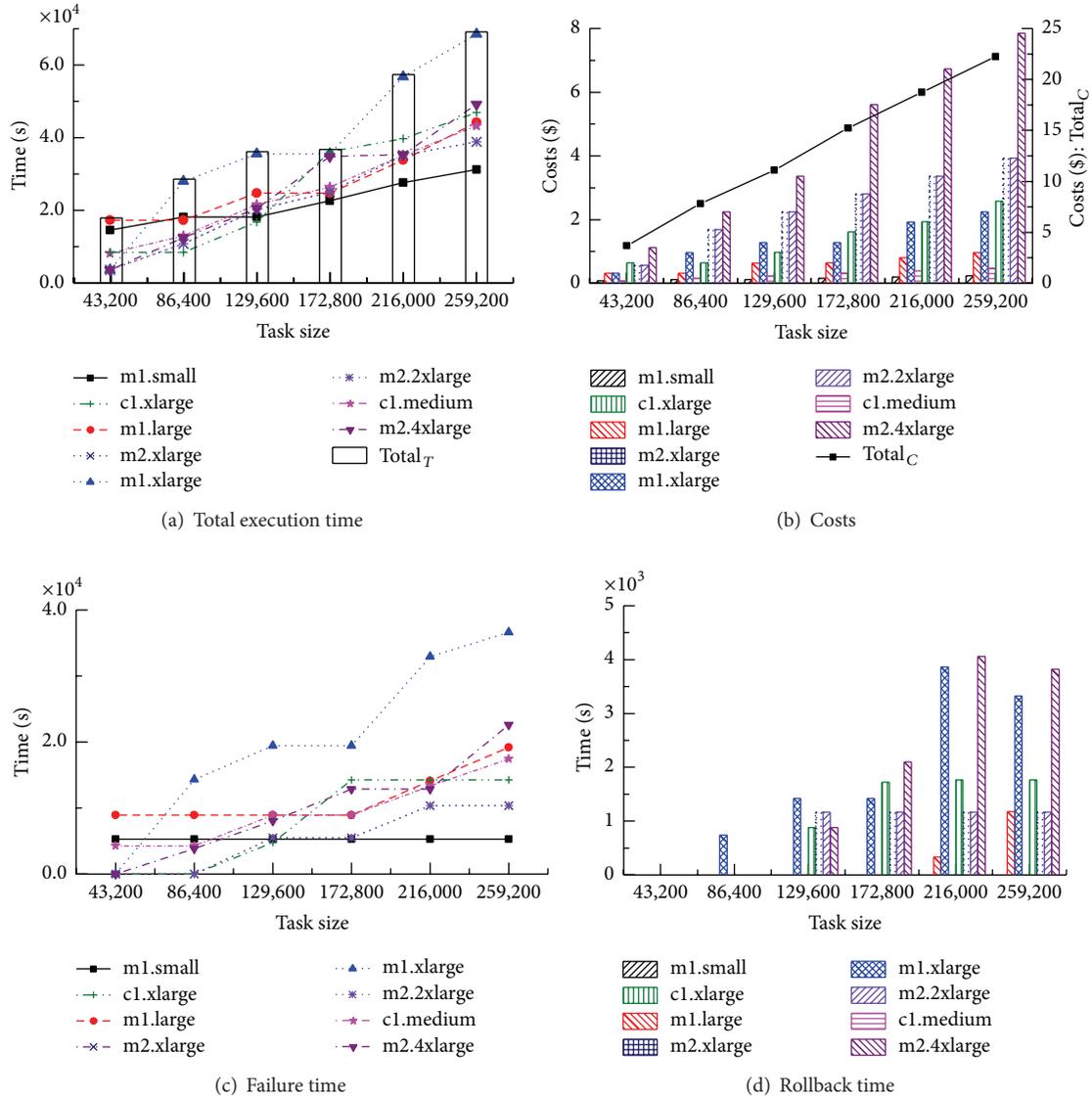


FIGURE 6: Simulation result in task distribution.

the total costs after applying our scheme decreased by an average of \$0.37 when compared to the cost before applying it. There are two facts deduced from these results. One is the increase of failure time. The other is the improvement of total execution time through an efficient task distribution. And the task execution loss was reduced when the out-of-bid situation occurred. In addition, we compare experiments to consider the execution time and costs.

Figure 8 shows the combined performance metric and the product of the total task execution time and cost. According to the task time interval, there is a little difference between the basic and the applying schemes, compared to each instance. In the figure, the basic scheme denotes the workflow product that applies only task distribution without considering a task processing rate. The applying scheme denotes the workflow product considering the task processing rate. The product of the basic scheme achieves performance improvements

in the average combined metric of 87.71% over the average product instance in each task time interval. The applying scheme achieves performance improvements in the average combined metric of 12.76%, compared to the basic scheme.

## 5. Conclusion

In this paper, we proposed a workflow scheduling technique considering task processing rate in unreliable cloud computing environments. The workflow scheduling scheme recalculates the task size based on task processing rate within the recalculated point. In addition, our previously proposed checkpoint scheme takes a checkpointing based on two kinds of thresholds: price and time. Our scheme reduces a failure time and an absolute time through the checkpoint scheme. The rollback time of our scheme can be less than that of

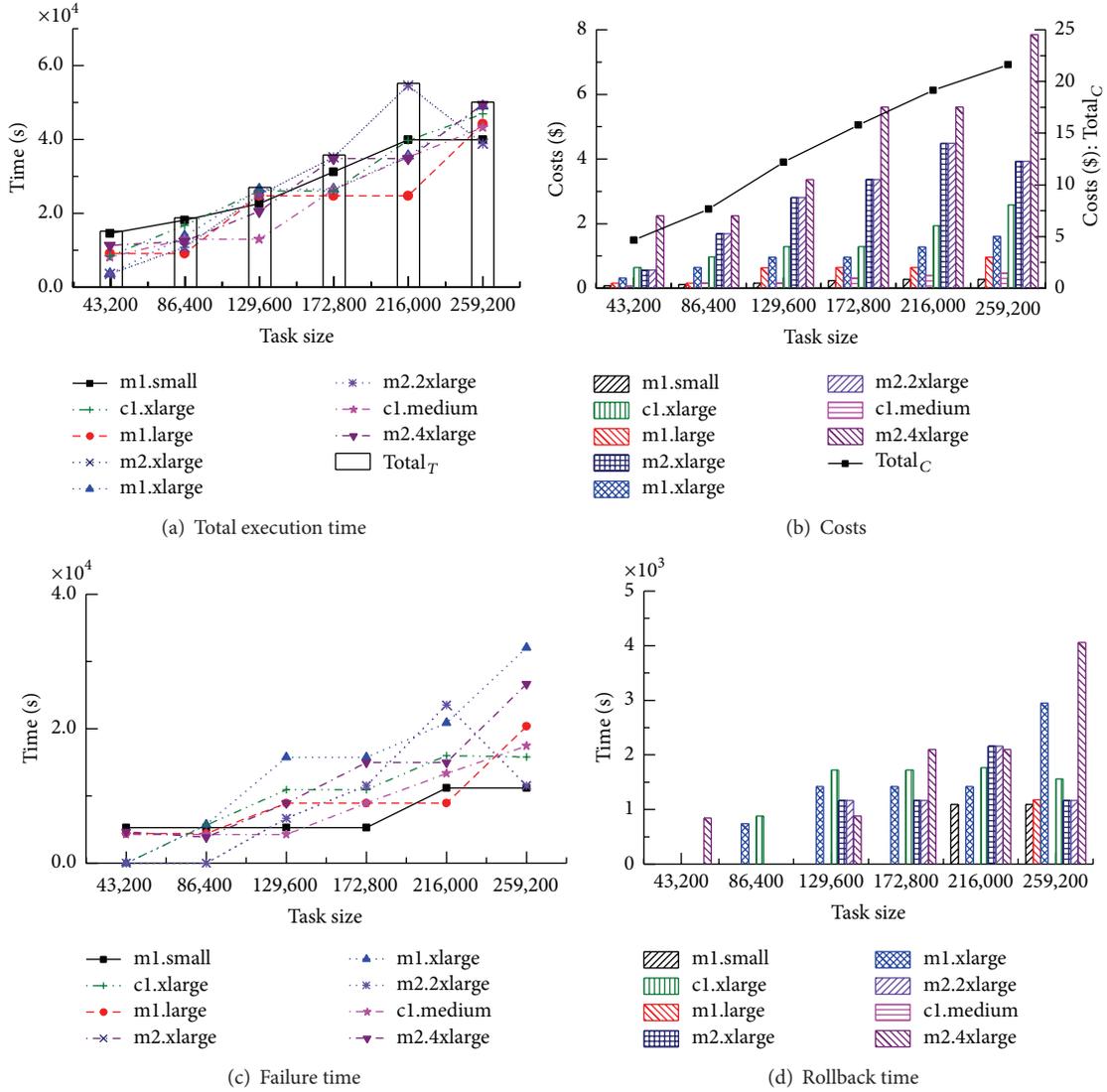


FIGURE 7: Simulation result in task distribution considering task processing rate.

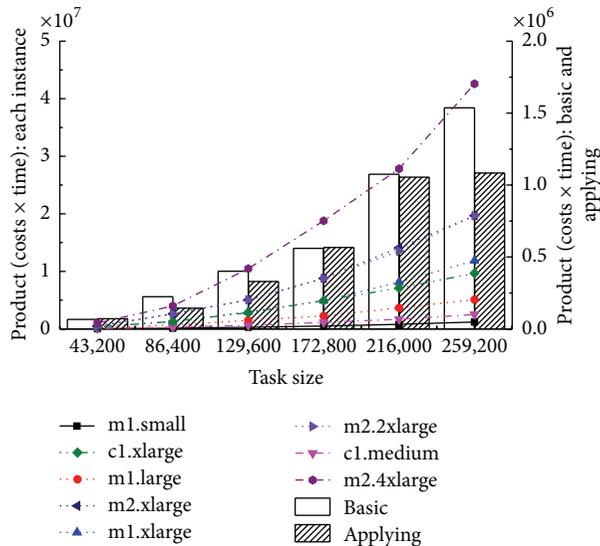


FIGURE 8: Comparison of combined metrics (total task execution time and cost).

the existing scheme without workflow because our scheme adaptively performs task distribution operation according to available instances. The simulation results showed that the average execution time in our scheme was improved by 17.8% after applying our proposed scheme as compared to before applying it. And our proposed scheme represented approximately the same cost as compared to before applying it. Other simulation results reveal that, compared to various instance types, our scheme achieves performance improvements in terms of an average combined metric of 12.76% over workflow scheme without considering task processing rate.

### 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 Korean government (MEST) (NRF-2012RIA2A2A02046684).

### References

- [1] *Elastic Compute Cloud (EC2)*, 2013, <http://aws.amazon.com/ec2>.
- [2] H. N. Van, F. D. Tran, and J.-M. Menaud, "SLA-aware virtual resource management for cloud infrastructures," in *Proceedings of the 9th IEEE International Conference on Computer and Information Technology (CIT '09)*, vol. 1, pp. 357–362, Xiamen, China, October 2009.
- [3] K. Mahajan, A. Makroo, and D. Dahiya, "Round robin with server affinity: a VM load balancing algorithm for cloud based infrastructure," *Journal of Information Processing System*, vol. 9, no. 3, pp. 379–394, 2013.
- [4] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: vision, hype, and reality for delivering IT services as computing utilities," in *Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC '08)*, pp. 5–13, Dalian, China, September 2008.
- [5] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Proceedings of the Grid Computing Environments Workshop (GCE '08)*, pp. 1–10, Austin, Tex, USA, November 2008.
- [6] M. M. Weng, T. K. Shih, and J. C. Hung, "A personal tutoring mechanism based on the cloud environment," *Journal of Convergence*, vol. 4, no. 3, pp. 37–44, 2013.
- [7] A. Følstad, K. Hornbæk, and P. Ulleberg, "Social design feedback: evaluations with users in online ad-hoc groups," *Human-Centric Computing and Information Sciences*, vol. 3, article 18, 2013.
- [8] *Amazon EC2 Spot Instances*, 2013, <http://aws.amazon.com/ec2/spot-instances/>.
- [9] SpotCloud, 2014, <http://www.spotcloud.com>.
- [10] S. Yi, J. Heo, Y. Cho, and J. Hong, "Taking point decision mechanism for page-level incremental checkpointing based on cost analysis of process execution time," *Journal of Information Science and Engineering*, vol. 23, no. 5, pp. 1325–1337, 2007.
- [11] S. Yi, D. Kondo, and A. Andrzejak, "Reducing costs of spot instances via checkpointing in the Amazon Elastic Compute Cloud," in *Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD '10)*, pp. 236–243, July 2010.
- [12] D. Jung, S. Chin, K. Chung, H. Yu, and J. Gil, "An efficient checkpointing scheme using price history of spot instances in cloud computing environment," in *Proceedings of the 8th IFIP International Conference on Network and Parallel Computing*, vol. 6985 of *Lecture Notes in Computer Science*, pp. 185–200, 2011.
- [13] H. Fernandez, M. Obrovac, and C. Tedeschi, "Decentralised multiple workflow scheduling via a chemically-coordinated shared space," Research Report RR-7925, Inria, 2012.
- [14] K. Liu, J. Chen, Y. Yang, and H. Jin, "A throughput maximization strategy for scheduling transaction-intensive workflows on SwinDeW-G," *Concurrency Computation Practice and Experience*, vol. 20, no. 15, pp. 1807–1820, 2008.
- [15] Cloud exchange, 2011, <http://cloudexchange.org>.

## Research Article

# Mathematical Modeling of a Multilayered Drift-Stabilization Method for Micro-UAVs Using Inertial Navigation Unit Sensor

Hyeok-June Jeong,<sup>1</sup> Myungwon Hwang,<sup>2</sup> Hanmin Jung,<sup>2</sup> and Young-guk Ha<sup>1</sup>

<sup>1</sup> Department of Computer Science & Engineering, Konkuk University, Neungdong-ro 120, Gwangjin-gu, Seoul 143-701, Republic of Korea

<sup>2</sup> Department of Computer Intelligence Research, Korea Institute of Science and Technology Information (KISTI), 245 Daehak-ro, Yuseong-gu, Daejeon 305-806, Republic of Korea

Correspondence should be addressed to Young-guk Ha; [ygha@konkuk.ac.kr](mailto:ygha@konkuk.ac.kr)

Received 19 February 2014; Accepted 6 May 2014; Published 22 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Hyeok-June Jeong 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.

This paper proposes a multilayered quadrotor control method that can move the quadrotor to the desired goal while resisting disturbance. The proposed control system is modular, convenient to design and verify, and easy to extend. It comprises three layers: a physical layer, a displacement control layer, and an attitude control layer. The displacement control layer considers the movement of the vehicle, while the attitude control layer controls its attitude. The physical layer deals with the physical operation of the vehicle. The two control layers use a mathematical method to provide minute step-by-step control. The proposed control system effectively combines the three layers to achieve drift stabilization.

## 1. Introduction

Unmanned aerial vehicles (UAVs) are expected to become a major part of the aviation industry as they can perform tasks such as traffic control, video recording, reconnaissance, and surveillance. Concomitant with developments in computer science, automatic control, sensors, and communications technologies, the quadrotor, in particular, is being evaluated as a suitable platform for small UAVs. Among its many advantages is the fact that it can move both vertically and horizontally, it can be made very small, and it can carry a variety of electrical devices.

A quadrotor flies by means of four propellers, which are controlled by an automatic system programmed in a micro-processor. This means that the control system has minute control over its flight. Consequently, it is relatively easier to fly a quadrotor than other aircrafts such as helicopters and airplanes.

However, controlling the desired behavior of the quadrotor is never easy. Because it flies in air, the quadrotor has to overcome inertia and undesirable wind disturbance.

Consequently, in order for the quadrotor to be useful, it has to have an effective control system.

Efforts have previously been made to solve this problem; however, this goal has not yet been definitively achieved. The control systems proposed thus far are more complicated than effective. Some researchers have proposed vision-based control systems; however, such systems are inefficient and heavy and therefore not suitable for small UAVs.

In this paper, we propose a quadrotor control system that uses the four motors and nine EOF-axis inertial navigation sensors, which measure the quadrotor's attitude and movement, to achieve minute control.

The control system operates on the basis of positioning; that is, it can move the quadrotor to any desired point while resisting external forces such as wind. Of course, drift stabilization is possible in any current position. In particular, the proposed control system is implemented in a modular form, so its design efficiency and performance can be easily verified.

The proposed controller significantly increases the quadrotor's stability and hovering performance and thereby facilitates the use of the quadrotor for various applications.

## 2. Related Work

Over the past decade, much research has been devoted to quadrotor control systems, a large portion of which has focused on the use of vision. Grzonka et al. [1] pioneered work in this area. Their research focused on indoor quadrotor flight control systems that are able to pilot the quadrotor in indoor spaces via real-time image processing [2]. However, even though their research is excellent, the stability of their control system is in doubt. Bills et al. [3] also proposed a control system that uses vision; however, their proposed system cannot guarantee stable quadrotor movement.

Romero et al. [4] proposed a control system that uses optical signals. The proposed system is stable because it only traces certain points. Gu et al. [5] focused their research on systems flying in formation, with each UAV designed hierarchically according to its role, such as flight leader or wingman. These systems have the advantage of stability but are limited to operating only in particular situations.

Zhang et al. [6] proposed a control system that combines vision with an IMU sensor. The system is stable; however, the efficiency of the control system is poor—it performs well for small vehicles but is too large for microquadrotors.

Consequently, in spite of research efforts expended to date, a suitable implantable control system for micro-UAVs has not been realized. We believe our proposed multilayered drift-stabilization method can provide an effective solution that resolves these problems.

## 3. Design of the Proposed Control System

In this section, we discuss the design of the proposed control system in terms of its layers and the algorithms used by each layer.

**3.1. The Combined Layers.** Our proposed multilayered control system has advantages in terms of its design, verifiability, and extensibility. These advantages result from the fact that each control layer is in charge of an assigned function. That is, each layer is considered to be abstracted.

The architecture of our proposed multilayered drift-stabilization control system is depicted in Figure 1. It comprises three layers. The first layer consists of physical components such as motor, frame, rotor, and battery. The rotation of the motor is controlled in accordance with a control value. The sensor calculates such parameters as the angular velocity and acceleration and passes the feedback values to the other layers.

The second layer, the “attitude control layer,” is in charge of the attitude angle, “roll, pitch, and yaw.” The attitude control layer receives attitude feedback and an objective angle. However, the layer does not care about how the actual angle is calculated. This layer is simply responsible for ensuring that the quadrotor is at the correct angle received.

Finally, the “displacement control layer” is in charge of the movement of the vehicle. This layer receives displacement feedback and control signals (reference) that it uses to calculate an appropriate angle to pass to the attitude control layer. The displacement control layer does not care how the

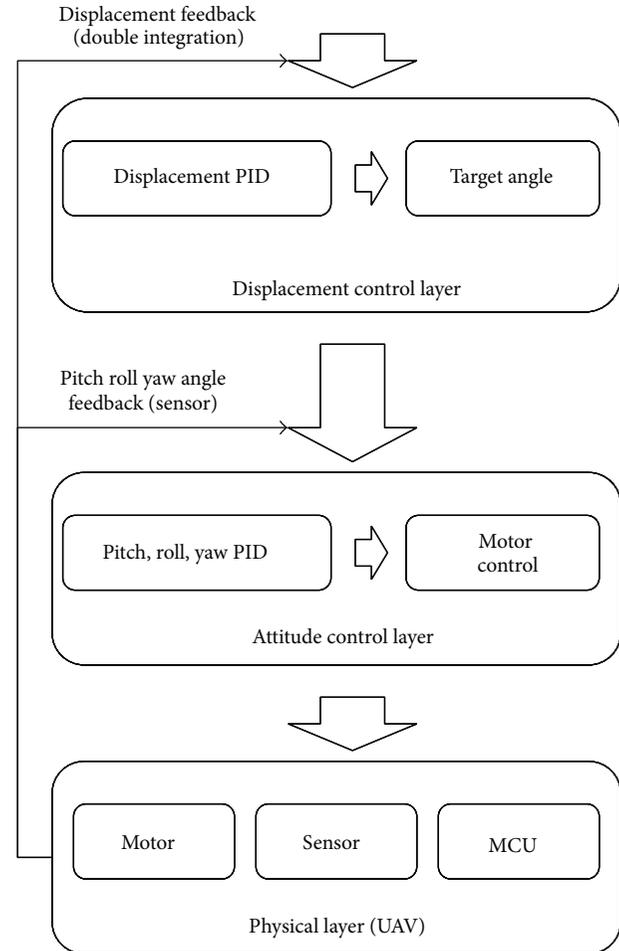


FIGURE 1: Architecture of the multilayered drift-stabilization control system.

angle of the vehicle is controlled. This is why we say that the control system design is abstracted.

Consequently, the proposed multilayered control system can be a powerful and convenient system, in which each section only fulfills its specified roles.

**3.2. The Physical Layer.** A quadrotor has four rotors and four motors on a rigid framework. A sensor is mounted at the center of the frame for accurate operations. Other components, such as circuit and batteries, are mounted in the same position as the sensor to ensure a balanced center of gravity.

Our proposed control system is designed for the shape shown in Figure 2 and takes into consideration the dynamics of this shape.

Despite the shape of the camera, the quadrotor is assumed to be symmetrical in quality and structure. The physical characteristics of the quadrotor are listed in Table 1.

In order to establish the dynamic model of the quadrotor, we can make the following general assumptions.

- (i) Gravity and resistance of the quadrotor are not affected by flight altitude and other factors.

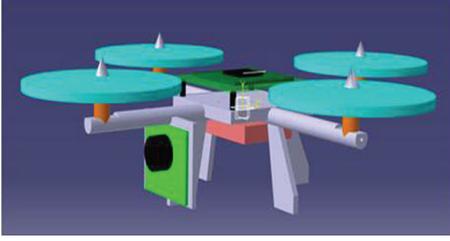


FIGURE 2: Physical appearance/shape of a quadrotor.

TABLE 1: Quadrotor physical parameters.

|                                         | Parameters | Value                                          |
|-----------------------------------------|------------|------------------------------------------------|
| Inertia around X-axis                   | $I_x$      | $7.5 \times 10^{-3} \text{ kg}\cdot\text{m}^2$ |
| Inertia around Y-axis                   | $I_y$      | $7.5 \times 10^{-3} \text{ kg}\cdot\text{m}^2$ |
| Inertia around Z-axis                   | $I_z$      | $1.5 \times 10^{-3} \text{ kg}\cdot\text{m}^2$ |
| Distance to the center of the quadrotor | $L$        | 0.23 m                                         |
| Mass of the quadrotor                   | $m$        | 0.65 kg                                        |
| Gravitational acceleration              | $g$        | $0.98 \text{ m}\cdot\text{s}^{-2}$             |

- (ii) Thrust in all directions is proportional to the square of the rotor speed.
- (iii) The quadrotor is a symmetrical rigid body.
- (iv) The origin of the inertial coordinate system is in the same position as the geometric center and the centroid of the quadrotor.

Two main effects are taken into consideration: generation of the thrust and the drag force. The thrust,  $T$ , produced by each motor is a force calculated as

$$F_i = \rho C_t A \omega_i^2 R^2 = k_t \omega_i^2, \quad (1)$$

where  $C_t$  is the thrust coefficient,  $\rho$  is the air density,  $A$  is the rotor disk area, and  $R$  is the blade radius. Further, the drag force is defined as

$$D_i = \frac{1}{2} \rho C_d v^2 = k_d v^2, \quad (2)$$

where  $D$  is the drag force,  $C_d$  is the drag force coefficient, and  $v$  is the speed of the quadrotor.

In the fixed coordinates of the body, the direct inputs are revolutions per minute (RPM) commands for the motors. The resultant outputs are  $Z$  direction thrusts in these coordinates.

However, the outputs under consideration are the attitude and position. To eliminate this gap, four control variables are defined as follows:

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} F_1 + F_2 + F_3 + F_4 \\ l(F_4 - F_2) \\ l(F_3 - F_1) \\ F_2 + F_4 - F_3 - F_1 \end{bmatrix} = \begin{bmatrix} k_t \sum_{i=1}^4 \omega_i^2 \\ k_t (\omega_4^2 - \omega_2^2) \\ k_t (\omega_3^2 - \omega_1^2) \\ k_d (\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix}, \quad (3)$$

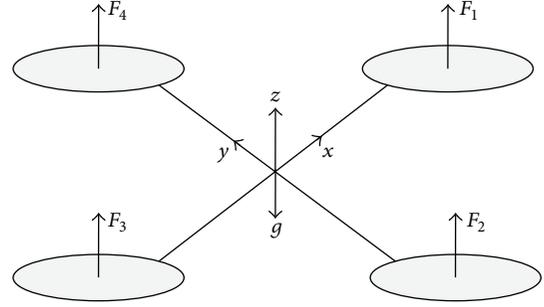


FIGURE 3: Definition of axis and rotor output.

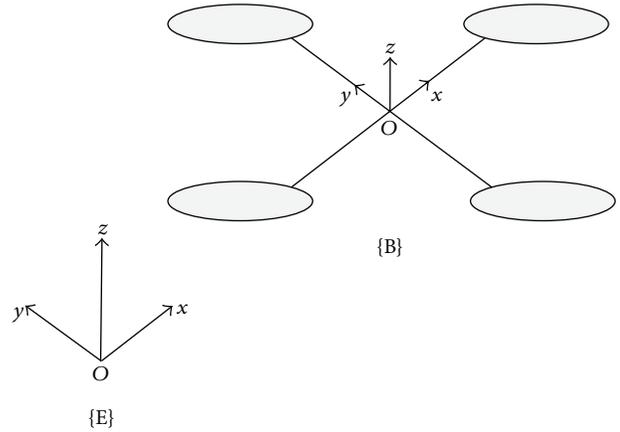


FIGURE 4: Model of the structure of the quadrotor.

where  $\omega_1, \omega_2, \omega_3$ , and  $\omega_4$  are the respective rotational speed of each rotor,  $F_1, F_2, F_3$ , and  $F_4$  are the lift forces of the motor's axis,  $l$  is the length of the quadrotor,  $U_1$  is the total lift force,  $U_2$  is the rolling moment,  $U_3$  is the pitching moment, and  $U_4$  is the yawing moment.

**3.3. The Attitude Control Layers.** Let us now derive a mathematical model for the quadrotor shown in Figure 2 (see Figure 4). But this layer is not our main interest. For that reason, this layer is designed using existing excellent research. Descriptions, expressions, sentences, and equations also are quoted in those papers (especially reference [7]).

The origin of the inertial coordinate system E is the initial position of the quadrotor. The positive direction of the OX axis is the designated heading of the quadrotor and is perpendicular to the horizontal plane.

This coordinate system is used to study the relative movement of ground and quadrotor. The OY axis is perpendicular to the OXZ plane. The quadrotor's spatial coordinates (X, Y, Z) can be obtained through the inertial coordinate system.

The origin of quadrotor coordinate system B ( $oxyz$ ) is the center of the quadrotor, and  $ox$  is parallel to the center connection of the front and rear rotors and the positive direction points to the front.  $oz$  is parallel to the center connection of the left and right rotors and the positive direction points to the right. The  $oy$  axis is perpendicular

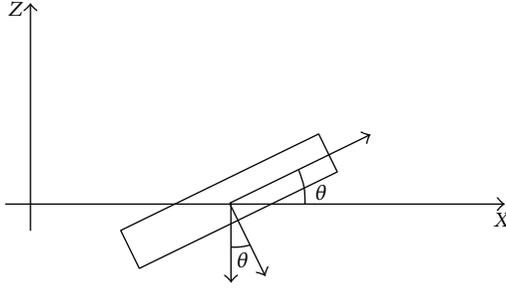


FIGURE 5: Directional diagram of the control system.

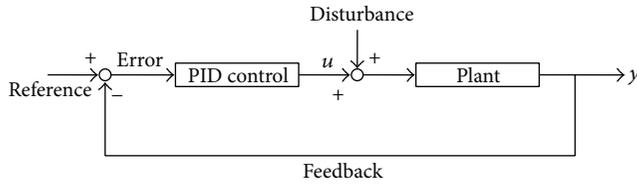


FIGURE 6: Block diagram of the PID feedback.

to the  $oxz$  plane; the positive direction is the direction conforming to the right hand rule.

These two coordinates can be converted to each other through transition matrix  $R$ .

Euler angles are defined as follows.

Pitch angle  $\theta$ : angle between the  $Z$ -axis and the projection of  $Oz$  in the  $OXY$  plane.

Yaw angle  $\varphi$ : angle between the  $X$ -axis and the projection of  $Ox$  in the  $OXY$  plane.

Roll angle  $\psi$ : angle between the  $Y$ -axis and the projection of  $Oy$  in the  $OXY$  plane.

Consequently, we can obtain the transition matrix  $R$ , which is from the quadrotor coordinate system to the inertial frame. Consider

$$\begin{aligned} R_x &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}, \\ R_y &= \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, \\ R_z &= \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (4)$$

With attitude angles defined as in Figure 2, the transformation matrix from the inertial coordinates to the fixed body coordinates is

$$\begin{aligned} R(\phi, \theta, \psi) &= R_x \cdot R_y \cdot R_z \\ &= \begin{bmatrix} \cos \psi \cos \phi & \cos \psi \sin \theta \sin \phi & \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\ \sin \psi \cos \theta & \sin \psi \sin \theta \sin \phi & \sin \psi \sin \theta \cos \phi - \sin \phi \cos \psi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix}. \end{aligned} \quad (5)$$

Then, we can obtain

$$\begin{aligned} F_x &= k_t \sum_{i=1}^4 \omega_i^2 (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi), \\ F_y &= k_t \sum_{i=1}^4 \omega_i^2 (\sin \psi \sin \theta \cos \phi + \cos \psi \sin \phi), \\ F_z &= k_t \sum_{i=1}^4 \omega_i^2 (\cos \phi \cos \phi). \end{aligned} \quad (6)$$

By Newton's second law of motion

$$\vec{F} = ma = m \frac{d\vec{V}}{dt}. \quad (7)$$

By Newton's second law, the dynamic equation of the quadrotor, the line motion equation can be obtained. It is defined as follows:

$$\begin{aligned} \ddot{x} &= \frac{(F_x - K_1 \dot{x})}{m} \\ &= \frac{k_t \sum_{i=1}^4 \omega_i^2 (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) - K_1 \dot{x}}{m}, \\ \ddot{y} &= \frac{(F_y - K_2 \dot{y})}{m} \\ &= \frac{k_t \sum_{i=1}^4 \omega_i^2 (\sin \psi \sin \theta \cos \phi + \cos \psi \sin \phi) - K_2 \dot{y}}{m}, \\ \ddot{z} &= \frac{(F_z - K_3 \dot{z} - mg)}{m} \\ &= \frac{k_t \sum_{i=1}^4 \omega_i^2 (\cos \phi \cos \phi) - K_3 \dot{z}}{m} - g, \end{aligned} \quad (8)$$

where,  $K_1 \dot{x}$ ,  $K_2 \dot{y}$ , and  $K_3 \dot{z}$  is the air resistance.

According to the relationship between Euler angle and angular velocity of the quadrotor, the following result can be obtained:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \psi \sin \theta \\ \dot{\theta} \cos \phi + \psi \sin \phi \cos \theta \\ -\dot{\theta} \sin \phi + \psi \cos \phi \cos \theta \end{bmatrix}. \quad (9)$$

Expressed with regard to  $\psi$ , we obtain

$$\begin{aligned} -\dot{\theta} &= \frac{r - \psi \cos \phi \cos \theta}{\sin \phi}, \\ \dot{\theta} &= \frac{q - \psi \sin \phi \cos \theta}{\cos \phi}, \\ \frac{\psi \cos \phi \cos \theta - r}{\sin \phi} &= \frac{q - \psi \sin \phi \cos \theta}{\cos \phi}. \end{aligned} \quad (10)$$

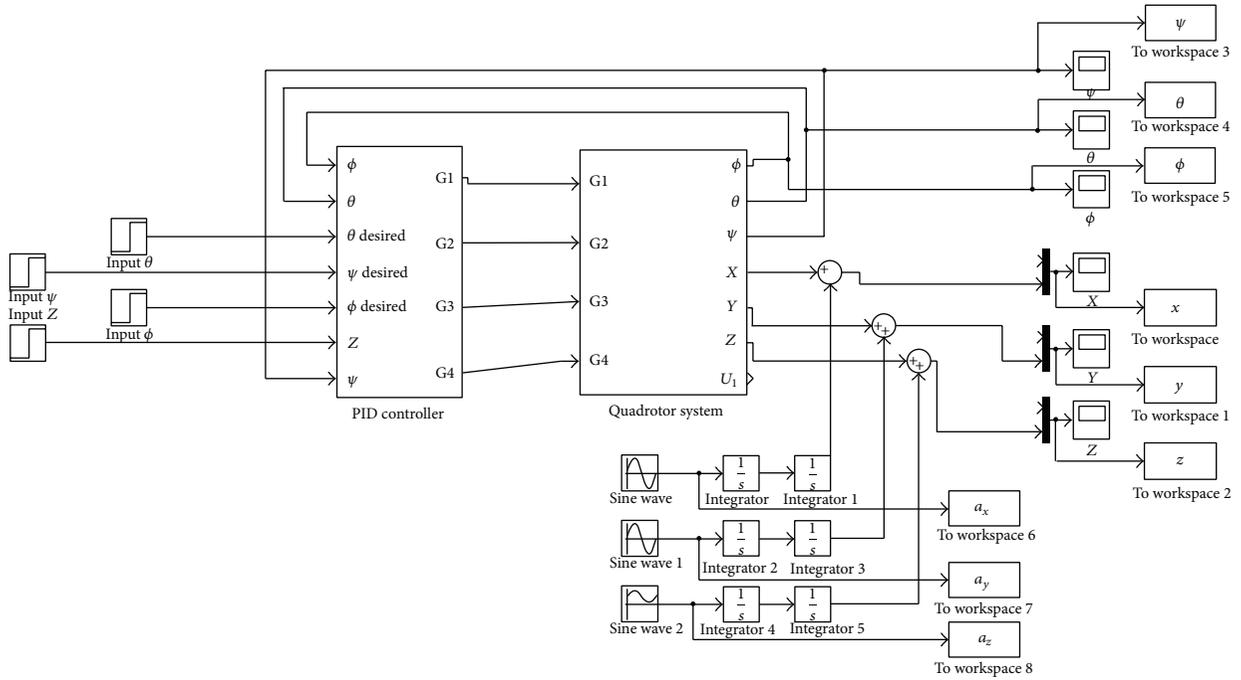


FIGURE 7: Implementation of the control system in MATLAB Simulink (layer 3 is excluded).

Then, this equation can be expressed as follows:

$$\psi (\cos^2 \phi \cos \theta + \sin^2 \phi \cos \theta) = r \cos \phi + q \sin \phi, \quad (11)$$

$$\psi = \frac{r \cos \phi + q \sin \phi}{\cos \theta}.$$

Finally, the following expression is obtained:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{p \cos \theta + q \sin \phi \sin \theta + r \cos \phi \sin \theta}{\cos \theta} \\ q \cos \phi + r \sin \phi \\ \frac{r \cos \phi + q \sin \phi}{\cos \theta} \end{bmatrix}. \quad (12)$$

The quadrotor was previously assumed to be symmetrical in quality and structure, so the inertia matrix. So it can be defined as a diagonal matrix:

$$I \begin{bmatrix} I_x & & \\ & I_y & \\ & & I_z \end{bmatrix}, \quad (13)$$

where  $I_{xx}$ ,  $I_{yy}$ , and  $I_{zz}$  are the rotary inertia around the X, Y, and Z axes, respectively.

By calculating the angular momentum, we can obtain the three axial components' angular motion equations of  $M$  in the quadrotor coordinate system:  $M_x$ ,  $M_y$ , and  $M_z$ . Consider

$$M_x = \dot{p}I_x + qr(I_z - I_y),$$

$$\dot{p} = \frac{M_x + qr(I_y - I_z)}{I_x}, \quad (14)$$

$$M_y = \dot{q}I_y + pr(I_x - I_z) + (p^2 - r^2)I_{xz},$$

$$M_y = \dot{q}I_y + pr(I_x - I_z), \quad (15)$$

$$\dot{q} = \frac{M_y + pr(I_z - I_x)}{I_y},$$

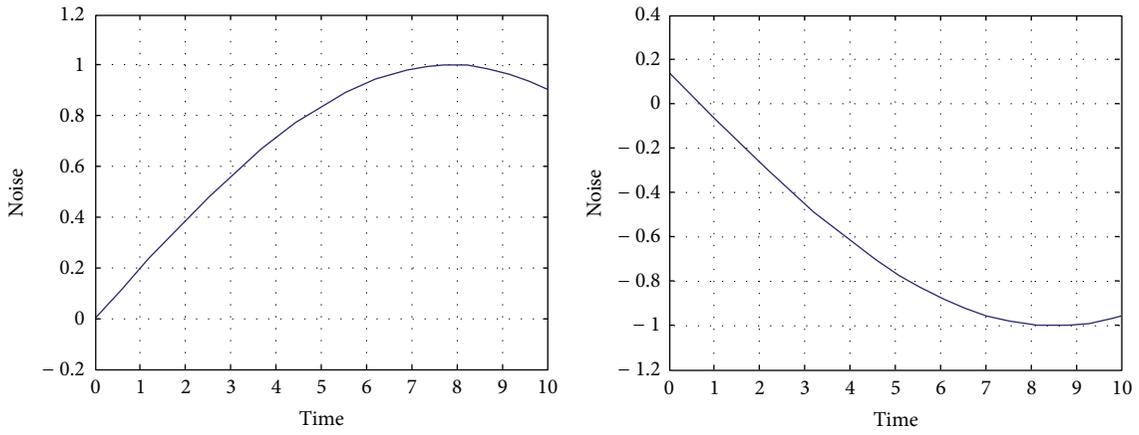
$$M_z = \dot{r}I_z - \dot{p}I_{xz} + pq(I_y - I_x) + qrI_{xz},$$

$$M_z = \dot{r}I_z + pq(I_y - I_x), \quad (16)$$

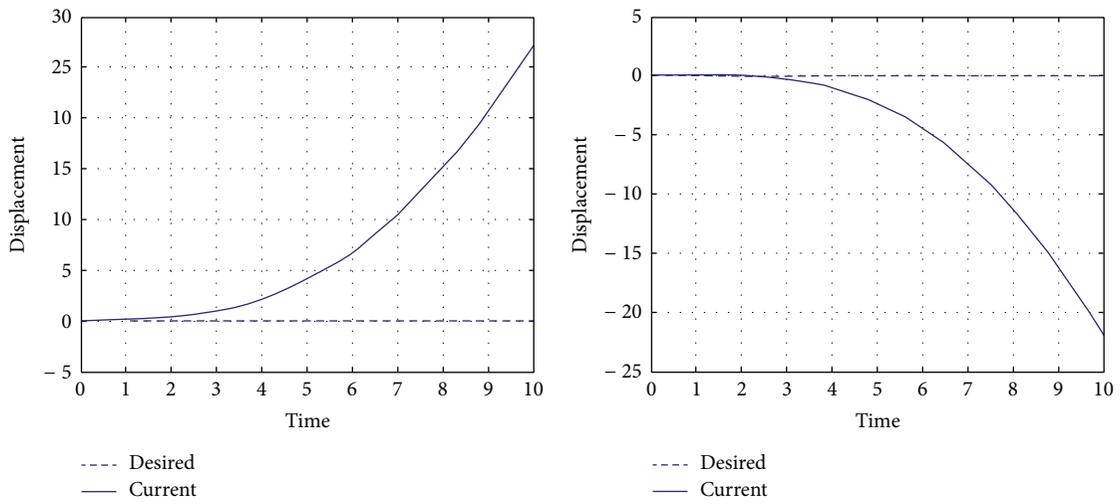
$$\dot{r} = \frac{M_z + pq(I_x - I_y)}{I_z}.$$

After simplification, the formula becomes

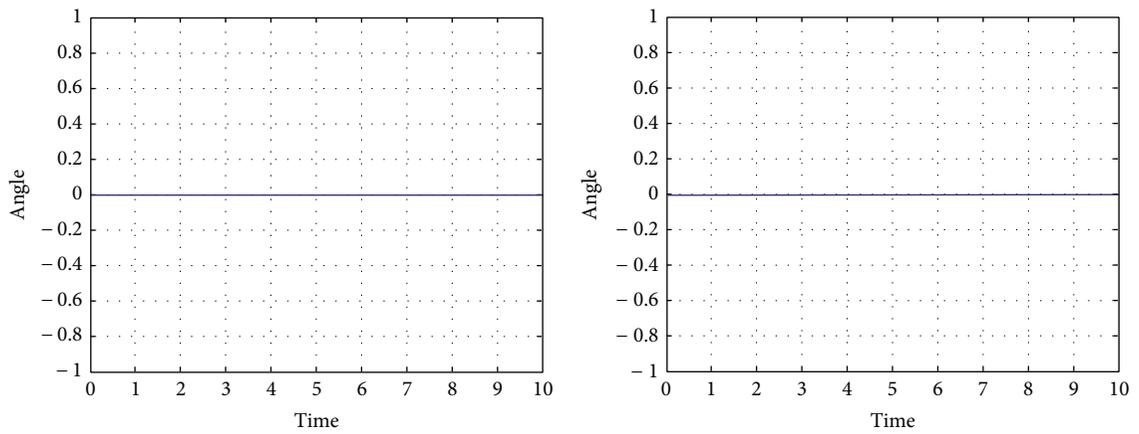
$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{M_x + qr(I_y - I_z)}{I_x} \\ \frac{M_y + pr(I_z - I_x)}{I_y} \\ \frac{M_z + pq(I_x - I_y)}{I_z} \end{bmatrix}. \quad (17)$$



(a) Continuous disturbance



(b) Position of quadrotor



(c) Attitude of quadrotor ( $\varphi, \theta$ )

FIGURE 8: Results from the partially simulated system (layer 3 is excluded).

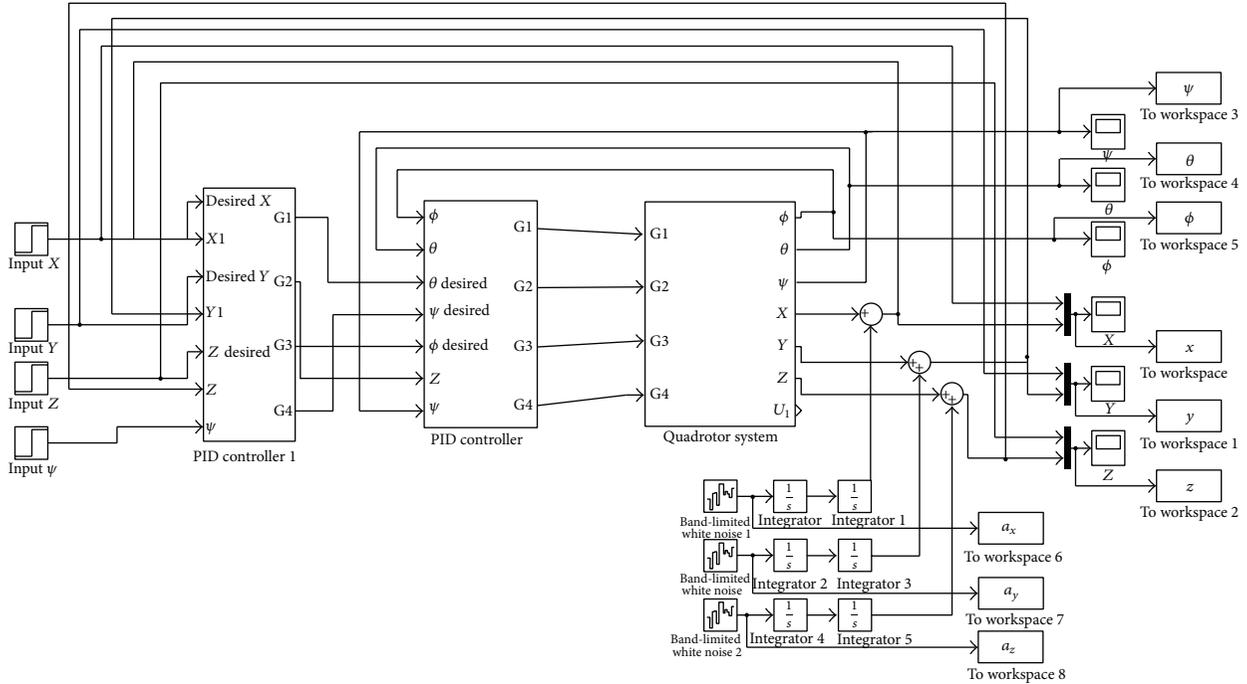


FIGURE 9: Implementation of the control system in MATLAB Simulink (full system).

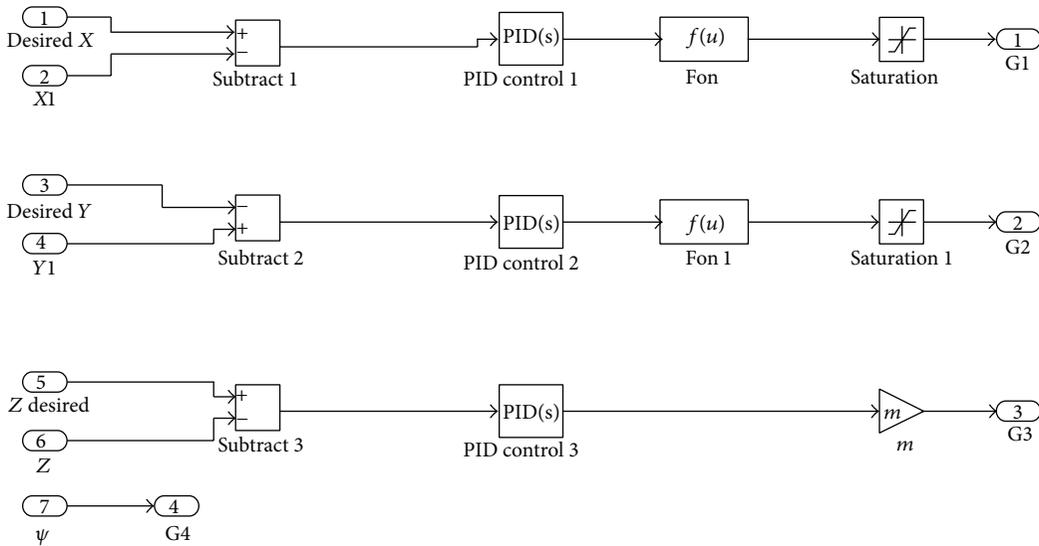


FIGURE 10: Inside of layer 3 block.

Equations (3)–(14) are mathematically considered state equations and can be implemented at the attitude control layer.

**3.4. The Displacement Control Layer.** The displacement control layer controls the movement of the quadrotor to the desired location while resisting disturbances. At this layer, the control system adjusts the attitude of the quadrotor to

make it move. That is, this layer receives the input value for position, which it then uses to calculate the proper angle for the quadrotor.

The algorithm is derived from situations such as those shown in Figure 5. The  $U_1$  is known, and the force of  $x$ -axis can be obtained by flowing equations. Consider

$$F_x = U_1 \sin \theta. \tag{18}$$

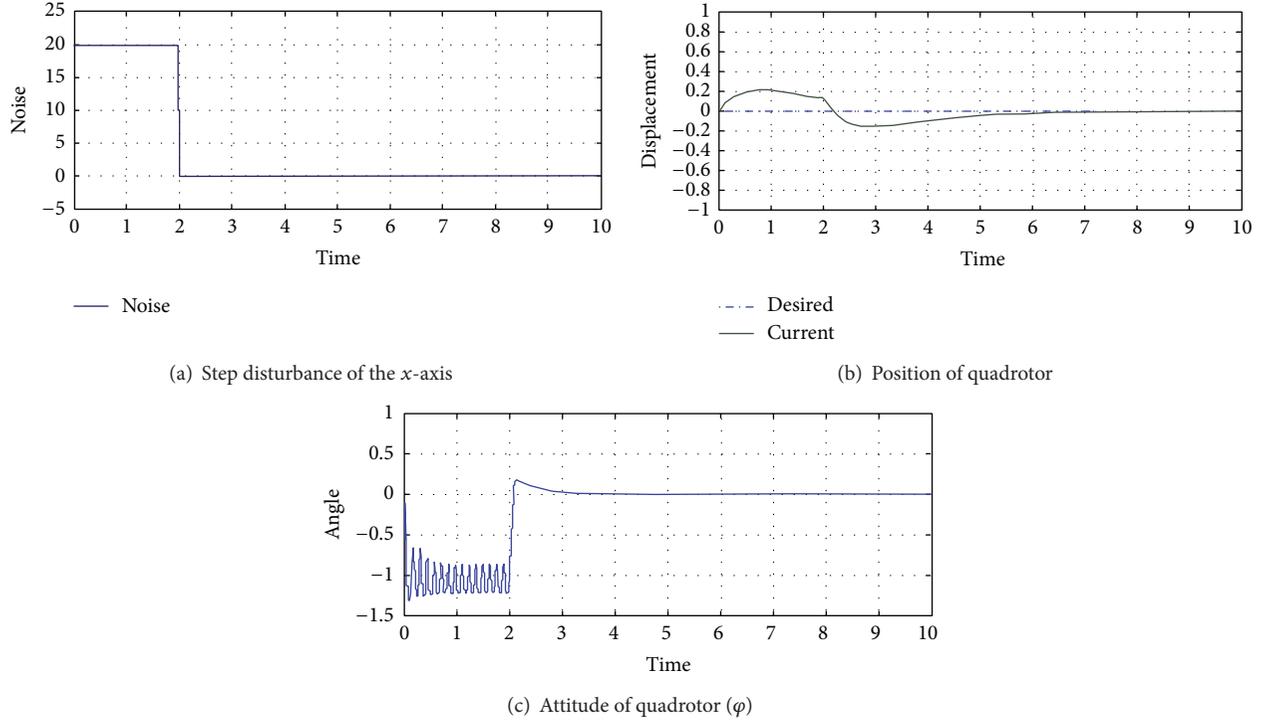


FIGURE 11: Results for the full system (disturbance only).

The quadrotor is tilted to the  $z$ -axis. This angle is defined as theta. The force is generated in the  $x$ -axis by the rotation of the rotor:

$$F_x = ma_x. \quad (19)$$

Equation (19) is Newton's second law on the  $x$ -axis. Consider

$$\sin \theta = \frac{a_x m}{U_1}, \quad (20)$$

$$\theta = \sin^{-1} \left( \frac{a_x m}{U_1} \right), \quad (21)$$

$$-1 \leq \frac{a_x m}{U_1} \leq 1. \quad (22)$$

Equation (21) can be obtained by substituting in (18) and (19). Then, an angle can be obtained for the desired acceleration. The respective domain and range of the sine function must be satisfied (22). The physical meaning of (21) is as follows.

- (i) In order to obtain a large acceleration, a large angle is necessary.
- (ii) A large mass requires a large angle.
- (iii) If the motor output is large, the desired angle is reduced (Figure 3).

Given the angle of the quadrotor, the respective acceleration can be derived easily; however, this system needs the

appropriate angle of the quadrotor's respective displacement. Therefore, we propose another equation derived from (18).

If the system is designed to simply resist the acceleration, it cannot achieve its desired target because of the value of the instantaneous acceleration. Our proposed control system needs to know the tendency of the movement due to acceleration. For this reason, the acceleration in (18) is replaced by a different function:

$$a_x \Rightarrow u_x(t). \quad (23)$$

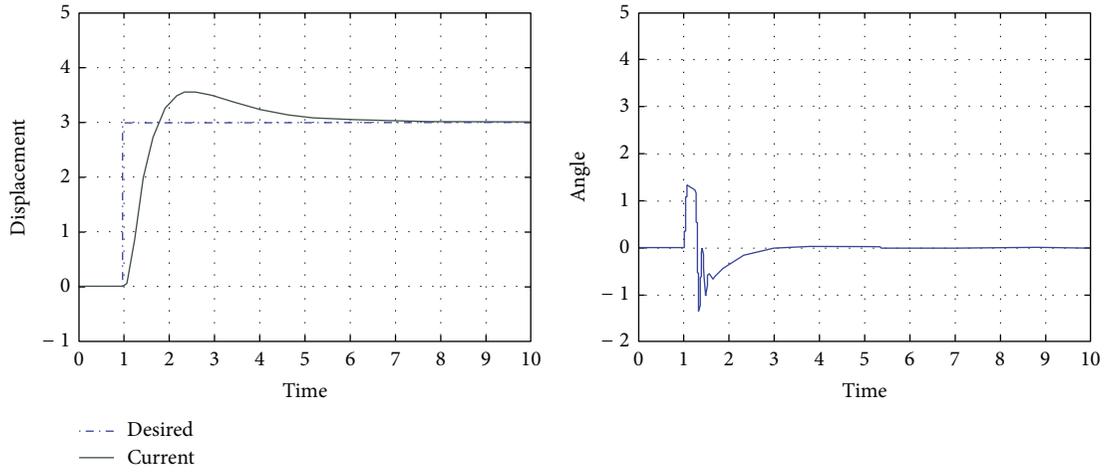
The conversion of expression (23) does not use the mathematical meaning of equal. The transform function is inferred from (18):

$$e_x(t) = d_{\text{ref}}(t) - d(t), \quad (24)$$

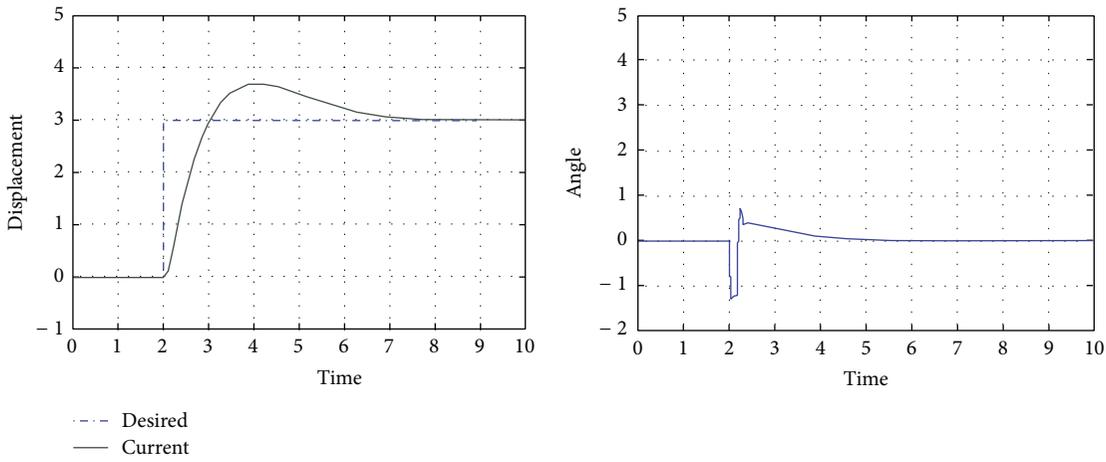
where  $d_{\text{ref}}$  is a desired location in the absolute coordinate system and  $d$  is the current position in the absolute coordinate system. The function (24) means the distance from the target point, in this sense called "error" in control engineering:

$$d(t) = d(t-1) + \iint a_{x_{\text{di}}}(t) dt + \iint \frac{(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) U_1}{m} dt, \quad (25)$$

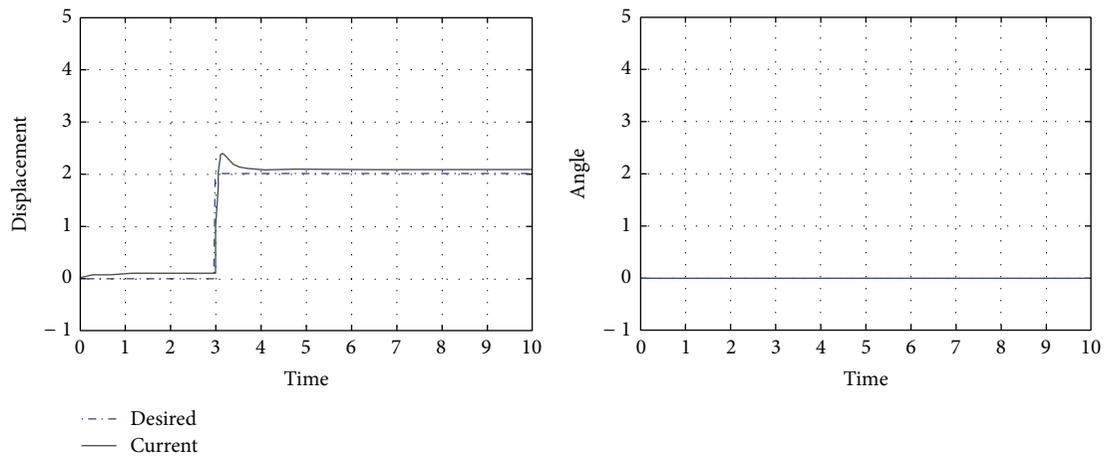
where  $d(t-1)$  is the previous position in the absolute coordinate system and  $a_{x_{\text{di}}}$  is the acceleration of the disturbance of the  $x$ -axis. This means that the current position is the sum of the previous position, the current attitude of the aircraft, and the acceleration of the disturbance.



(a) Position of quadrotor (x-axis)



(b) Position of quadrotor (y-axis)



(c) Position of quadrotor (z-axis)

FIGURE 12: Tracking displacement without noise.

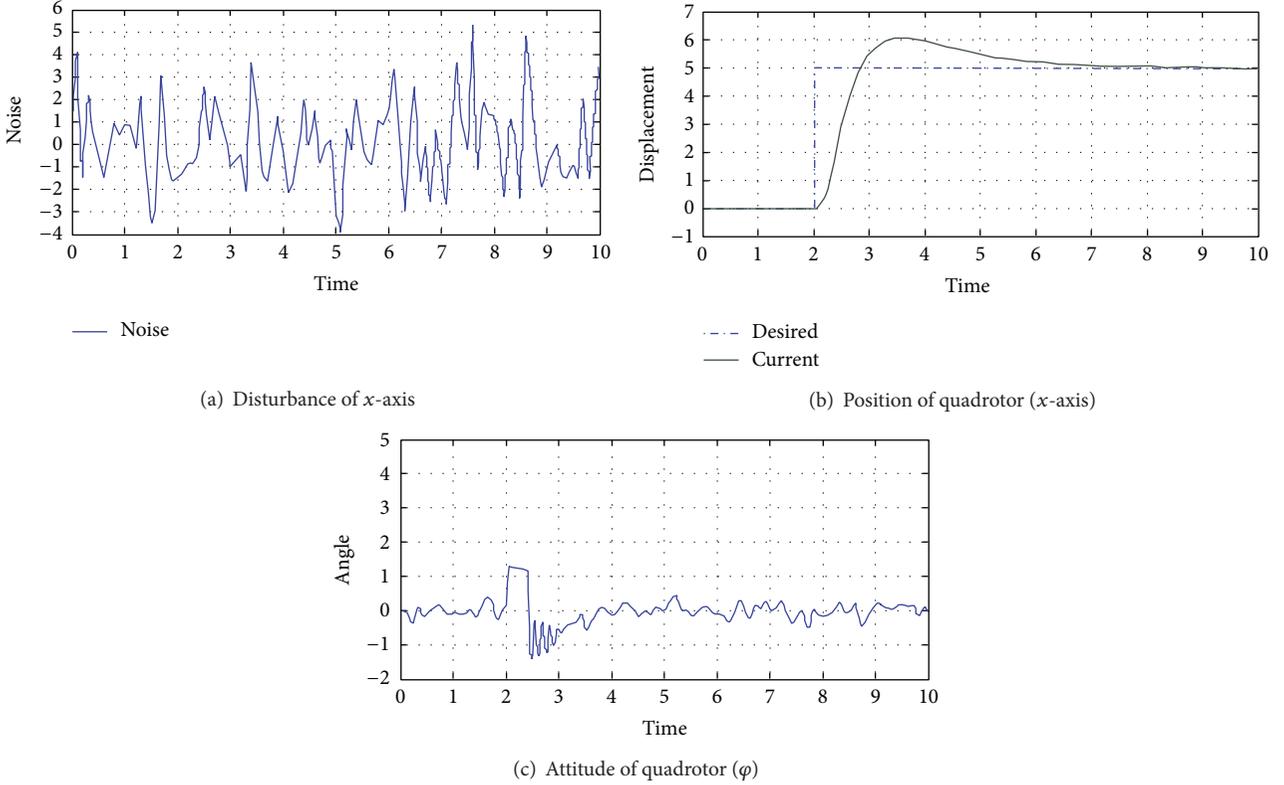


FIGURE 13: Tracking displacement with noise.

That is, the transform function refers to the error value in the PID feedback control shown in Figure 6, and the feedback control system can minimize the error by adjusting the process control outputs. Consider

$$u(t) = K_p e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt}, \quad (26)$$

where  $\tau$  is a variable of integration and takes values from time zero to the present time,  $t$ .  $K_p$ ,  $T_i$ , and  $T_d$  signify the gain, a tuning parameter. Consider

$$\theta = \sin^{-1} \left( \frac{u_x(t) m}{U_1} \right), \quad (27)$$

$$-\sin \theta_{\min} \leq \frac{u_x(t) m}{U_1} \leq \sin \theta_{\max}. \quad (28)$$

The appropriate angle can be calculated using (28). The physical meaning of this expression is as follows (Figure 7, left block).

- (i) A large mass requires a large angle.
- (ii) If the motor output is large, the desired angle is reduced.
- (iii) The larger the control signal, the more the angle tilts.

The displacement control layer enables the quadrotor to move to the desired point while resisting disturbance.

TABLE 2: Control parameter values.

|          | Parameters | Value |
|----------|------------|-------|
| $P$ term | $P$        | 1.2   |
| $I$ term | $I$        | 0.1   |
| $D$ term | $D$        | 0.4   |

Invariably, it also makes it possible for the quadrotor to hover. However, this control layer calculates only the angle for the respective distance error. In other words, this layer is designed to faithfully conduct an indigenous mission.

#### 4. Implementation and Experimentation

We implemented our proposed control system in MATLAB Simulink and conducted several simulations (Figure 7). The experimental values obtained are listed in Tables 1 and 2.

Optimization is not within the scope of this paper. As a result, approximate values were used. To verify our proposed multilayered control system, only layers 1 and 2 were implemented in the first set of experiments.

In Figure 7, the left block is the attitude control layer and the right block is the physical layer. The remaining small blocks are the input and output gates.

Figure 8 shows the results of the first set of experiments. As expected, in situations where there is a continuous disturbance, the quadrotor is able to maintain its attitude.

However, it can be seen that the quadrotor is being gradually pushed by the disturbance. The quadrotor control system is intended to give results such as those shown in Figure 8.

In the second set of experiments, we added layer 3 to the system.

The left block is the displacement control layer. It receives position feedback and outputs the appropriate angle (Figure 9).

Figure 10 shows the inside block that implements (26) and (27). With the addition of layer 3 to the system, the results depicted in the figure were obtained.

It can be seen in Figure 11 that the attitude changes to resist the disturbance. This appropriate change in the attitude enables the quadrotor to have drift stabilization.

The control system knows the acceleration due to the quadrotor control position; thus, control is possible. Figure 12 shows that it is moved to a point that maintains its objective.

Figure 13 shows that it moves to a point to maintain its objective even though the disturbance is very irregular. When there are large angular changes it is designed to move to a certain point, whereas for small angular changes it is designed to resist the disturbance.

The final experimental results show that our proposed multilayered drift-stabilization control system can be a useful and powerful solution for microquadrotors.

## 5. Conclusion

The design and implementation of mathematical modeling of our multilayered drift-stabilization method were successful. Each of the layers is designed separately and drift stabilization is achieved by cooperation with each layer.

Our proposed system provides powerful solutions for some of the problems that have not yet been solved. The system can be mounted on small vehicles and can make the quadrotor move to its objective even when there are irregular disturbances. The proposed control system has good performance and is easily expandable. Therefore, it is a useful micro-UAV control system.

However, this control system presently has an integration error. In order to overcome this limitation, we plan to implement a process that corrects it using vision. The system also lacks optimization, particularly in the PID term. Therefore, we hope to conduct further studies from which we can present optimized results that further enhance the usefulness of our proposed control system.

## Conflict of Interests

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

## Acknowledgment

This research was supported by the Basic Science Research Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (Grant no. 2012006817).

## References

- [1] S. Grzonka, G. Grisetti, and W. Burgard, "A fully autonomous indoor quadrotor," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 90–100, 2012.
- [2] S. G. Fowers, D. Lee, B. J. Tippetts, K. D. Lillywhite, A. W. Dennis, and J. K. Archibald, "Vision aided stabilization and the development of a quad-rotor micro UAV," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '07)*, pp. 143–148, Jacksonville, Fla, USA, June 2007.
- [3] C. Bills, J. Chen, and A. Saxena, "Autonomous MAV flight in indoor environments using single image perspective cues," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 5776–5783, Shanghai, China, May 2011.
- [4] H. Romero, S. Salazar, and R. Lozano, "Real-time stabilization of an eight-rotor UAV using optical flow," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 809–817, 2009.
- [5] D. L. Gu, G. Pei, H. Ly, M. Gerla, and X. Hong, "Hierarchical routing for multi-layer ad-hoc wireless networks with UAVs," in *Proceedings of the 21st Century Military Communications Conference (MILCOM '00)*, pp. 310–314, October 2000.
- [6] T. Zhang, Y. Kang, M. Achtelik, K. Kühnlenz, and M. Buss, "Autonomous hovering of a vision/IMU guided quadrotor," in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '09)*, pp. 2870–2875, Changchun, China, August 2009.
- [7] J. Li and Y. Li, "Dynamic analysis and PID control for a quadrotor," in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '11)*, pp. 573–578, College of Mechanical and Electrical Engineering China Jiliang University, August 2011.

## Research Article

# Towards Self-Awareness Privacy Protection for Internet of Things Data Collection

**Kok-Seng Wong and Myung Ho Kim**

*School of Computer Science and Engineering, Soongsil University, Information Science Building, Sangdo-dong, Dongjak-gu, Seoul 156-743, Republic of Korea*

Correspondence should be addressed to Myung Ho Kim; [kmh@ssu.ac.kr](mailto:kmh@ssu.ac.kr)

Received 10 February 2014; Accepted 6 May 2014; Published 19 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 K.-S. Wong and M. H. Kim. 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) is now an emerging global Internet-based information architecture used to facilitate the exchange of goods and services. IoT-related applications are aiming to bring technology to people anytime and anywhere, with any device. However, the use of IoT raises a privacy concern because data will be collected automatically from the network devices and objects which are embedded with IoT technologies. In the current applications, data collector is a dominant player who enforces the secure protocol that cannot be verified by the data owners. In view of this, some of the respondents might refuse to contribute their personal data or submit inaccurate data. In this paper, we study a self-awareness data collection protocol to raise the confidence of the respondents when submitting their personal data to the data collector. Our self-awareness protocol requires each respondent to help others in preserving his privacy. The communication (respondents and data collector) and collaboration (among respondents) in our solution will be performed automatically.

## 1. Introduction

The Internet of Things (IoT) is now an emerging global Internet-based information architecture used to facilitate the exchange of goods and services. The concept of IoT is to allow living objects (humans or animals), devices (sensor), or object with embedded technologies to automatically transfer data over communication networks (wired or wireless networks) without human-to-human or human-to-computer interaction. IoT aims to utilize and extend the benefits of Internet such as always-on, data sharing, and remote access capabilities [1].

IoT enables data collection in every aspect of our life. Data collected from smart metering application allows the utility provider to analyze and improve its services. Also, these data can help the user to be aware of their energy consumptions and possible energy saving strategies. In an underwater environment, smart meter is particularly important because information can be detected, gathered, and sent to the sensor [2].

Let us consider the following scenario. A practitioner (data collector) would like to collect medical data from his patients (respondents) with implanted medical devices. Since medical data are highly sensitive information, respondents must be aware of the data to be collected. There are two main paradigms to protect the patient's privacy in this scenario. The first paradigm relies on the respondent's trust in the data collector while the second paradigm depends on the respondent's anonymity. If the respondents do not have confidence in the data collector, they may refuse to submit data or provide inaccurate data to the agency. If the submitted data from the respondents are not genuine, we can predict that the data collector will face the data utility problem because the analyzed results based on the collected data will not be accurate. In the second paradigm, we should prevent the reidentification problem. For instance, if the collected data are used for research purposes, the data collector should not be able to link any of the collected data to the real identity of any patient.

*1.1. Challenges of IoT.* Wireless sensor networks have been revolutionized by creating significant impact throughout the society [3]. Advances in wireless communication technology (e.g., efficient resource management [4] and performance improvement [5] in wireless network) enable the development and implementation of IoT applications. IoT-related applications include traffic congestion detection and waste management in smart cities, remote diagnostics in patients' surveillance system (e.g., Ubiquitous healthcare [6, 7]), and storage condition monitoring in supply chain control.

Along with potential benefits offered, the usage of IoT also raises some privacy concerns to the data owners. In particular, real-time data collection and data analysis in IoT applications may compromise the privacy of data owner. In practical, new data arrive continuously and up-to-date data should be used for analysis. The data collected at different times allows malicious providers to learn extra knowledge by cross-examining the data within a targeted timeframe. Therefore, a secure and privacy aware protocol should be implemented in IoT when data are collected automatically. Some new security and privacy challenges can be found in [8].

The development of radio frequency identification (RFID) technologies and the advances of network communication technologies motivate the forming of IoT [9]. Physical objects called u-things which are embedded or connected to communication networks, sensors, and computers are commonly found in our daily life [10]. In the context of IoT, u-things should be able to act automatically (e.g., autodetection and data transfer) and adaptively. The construction of smart u-things involves the following 7 challenges [11, 12]:

- (i) surrounding situations (context),
- (ii) users' needs,
- (iii) things' relations,
- (iv) common knowledge,
- (v) self-awareness,
- (vi) looped decisions,
- (vii) ubiquitous safety (UbiSafe).

The ultimate goal of any ubiquitous intelligence is to make the u-things behave trustworthily in both other-aware and self-aware manners to some degrees and circumstances [13]. Therefore, it is important to design a self-awareness protocol to help data owners to protect their privacy.

In this paper, we will focus on the self-awareness challenge. In particular, we design a self-awareness protocol to increase the confidence of the data owner when the smart u-things automatically submit their data to the data collector.

*1.2. Problem Statement.* There are two challenges we aim to address in this work. Firstly, we want to protect the identity of each data owner from the data collector before and after the data collection process. Secondly, and more importantly, we want to guarantee the usefulness of the collected data by increasing the confidence of data owner.

The first challenge can be solved by using anonymity technology such as the onion routing (Tor) [14], anonymous proxy server [15], and mix network [16, 17]. These technologies are still under active investigation and their focuses are mainly on network traffic analysis, anonymous communication channel, and private information retrieval. Since our aim in this paper is not to design any of the specific anonymity technology, we refer readers to [15, 18] for the usage of these technologies.

The second challenge requires each respondent to help others in order to preserve his own privacy. This idea is motivated by the coprivacy concept in [19, 20]. Coprivacy (or cooperative privacy) considers the best option for a party to achieve his privacy protection is to help another party in achieving her privacy. The formal definition of coprivacy and its generalizations can be found in [19].

*1.3. Our Contributions.* In this paper, we propose a self-awareness protocol to facilitate the data collection in IoT-related applications. Instead of placing full trust on the utility provider (data collector), we allow each data owner (respondent) to learn the protection level provided by the data collector before the data submission process. We summarize our contributions as follows.

- (i) We propose a privacy preserved approach to enable the respondents to learn about the anonymous protection level they will receive from the data collector before the data submission.
- (ii) Our notion of self-awareness protection can be used to increase the confidence of respondents in the data collection process. Hence, respondents will feel comfortable to submit their genuine data while the data collector can ensure the usefulness of the collected data.

*1.4. Organization.* The rest of this chapter is organized as follows. The background and related work for this research are presented in Section 2. We describe the technical preliminaries of our solution in Section 3. We present our solution in Section 4 followed by analysis of correctness, privacy, efficiency, and discussion in Section 5. Our conclusion is in Section 6.

## 2. Background and Related Work

*2.1. Privacy Paradigm in IoT.* In 1973, the United States Department of Health, Education, and Welfare proposed Fair Information Practice Principles (FIPPs) as the guideline to assure fair practice and adequate data privacy protection. In particular, the guideline aims to protect the consumer rights such as how online entities should collect and use the personal data [21]. Five principles of FIPPs are as follows [22].

- (1) There must be no personal data record-keeping systems whose very existence is secret.
- (2) There must be a way for a person to find out what information about the person is in a record and how it is used.

- (3) There must be a way for a person to prevent information about the person that was obtained for one purpose from being used or made available for other purposes without the person's consent.
- (4) There must be a way for a person to correct or amend a record of identifiable information about the person.
- (5) Any organization creating, maintaining, using, or disseminating records of identifiable personal data must assure the reliability of the data for their intended use and must take precautions to prevent misuses of the data.

Based on the above principles, we now analyze the privacy protection in current IoT. Since data are collected automatically, it is hard for the data owners to ensure that their privacy can be protected. In most cases, utility providers will design a series of mechanisms to guarantee the privacy protection of the collected data. However, we found that data owners are generally not able to verify those mechanisms offered by the provider. Therefore, a self-awareness protocol should be available for automatic data collection process.

*2.2. Anonymous Data Collection.* In general, online data collection is a process which involves collaboration between a trusted party (data collector) and a number of data owners (respondents). Due to concerns regarding privacy, respondents might refuse to contribute their personal data or submit inaccurate data to the data collector. Therefore, the data collector needs to ensure the privacy of data submitted through a series of secure mechanisms. However, the protection level provided by the data collector is hard to be verified by the respondents.

Often, data collected from the respondents will be used for research or data analysis. The release of the collected data causes a privacy issue in data publishing, in particular, when it involves the republication of the same data in a given period [23]. There are two settings that can be observed when the data is released to the data recipient. If the data recipient is a third party, data must be released in an anonymous form without compromising the privacy of the respondents. Let us consider a scenario where a hospital (data collector) wishes to publish patients' records to a research institute (data recipient) for data analysis. In a common practice, all the explicit personal identity information (PII) such as name and social security number will be removed from the original dataset before it is released to the data recipient. However, removing PII does not preserve privacy.

Data anonymization is an interesting solution to protect the privacy of the respondents for this setting. Sweeney proposed  $k$ -anonymity model to address the linking attack [24]. The concept of  $k$ -anonymity [25] is such that each released data is indistinct from at least  $(k-1)$  other data. However,  $k$ -anonymity is found vulnerable against background knowledge attacks by Machanavajhala et al. [26].

In the literature, techniques such as  $(\alpha, k)$ -anonymity [27, 28],  $l$ -diversity [26], and  $t$ -closeness [29] have been proposed to enhance the  $k$ -anonymity model. We note that these techniques assumed that  $k$ -anonymity has been achieved

in the first place before applying additional techniques to enhance the anonymous protection of the released data. For instance,  $(\alpha, k)$ -anonymity model assumed that all the released data adhere to  $k$ -anonymity. In addition, it requires that the frequency of the sensitive value in any quasi-identifier is less than  $\alpha$  after the anonymization [27]. In the  $l$ -diversity model, the sensitive attribute in the  $k$ -anonymous table is well represented by  $l$  values such that each sensitive value is at most  $1/l$ . A survey of recent attacks and privacy models in data publishing can be found in [30].

In this paper, we consider the second setting where the data analysis is performed by the data collector. This scenario is more complex to deal with because the data collector has the full access to all raw data from the respondents. Therefore, we need to design a protocol to increase the confidence of the respondents before they submit their records to the data collector. In other words, respondents are aware of the protection level they received from the data collector after the data submission.

### 3. Related Works

Various self-oriented privacy protections have been proposed in the literature. Self-enforcing privacy (SEP) for e-polling was proposed in [31]. The idea of SEP is to enforce the pollster to protect the respondents' privacy by allowing the respondents to trace their data after the submission. If the pollster releases the poll results, the respondents can indict the pollster by using the evidence they obtained during the data collection process. A fair indictment scheme for SEP can be found in [32].

The most related research to our work in this paper is the respondent-defined privacy protection (RDPP) for anonymous data collection proposed in [33]. The basic idea of RDPP is to allow the respondents to specify the level of protection they require before providing any data to the data collector. For instance, a number of respondents (minimum threshold) must satisfy the constraint chosen by the respondent  $i$  before he agrees to submit the data. In their protocol, respondents are aware of the minimum level of privacy protection they will receive before submitting their dataset to the data collector. Instead of relying on the data collector to guarantee the privacy protection, the respondents are free to define their preferred protection level.

In this paper, we do not consider indictment for our protocol because the data analysis is done by the data collector. Instead of allowing the respondents to freely define their own privacies, we assume that respondents are willing to submit their data if the protection level offered by the data collector can be verified by them.

### 4. Technical Preliminaries

*4.1. Homomorphic Encryption Scheme.* We use homomorphic encryption scheme (i.e., Paillier [34]) as our primary cryptographic tool. Let  $\text{Enc}_{\text{pk}}(m)$  denote the encryption of  $m$  with the public key,  $\text{pk}$ . Given two ciphertexts,  $\text{Enc}_{\text{pk}}(m_1)$  and  $\text{Enc}_{\text{pk}}(m_2)$ , there exists an efficient algorithm  $+_h$  to compute

TABLE 1: Sample medical dataset.

| Patient | Gender | Age | Zip   | Disease       |
|---------|--------|-----|-------|---------------|
| Bob     | Male   | 15  | 27892 | Flu           |
| Sam     | Male   | 13  | 27886 | Heart disease |

$\text{Enc}_{\text{pk}}(m_1 + m_2)$ . This additive property can be performed without the decryption key.

*4.2. Definitions.* Let us assume that there are  $n$  respondents  $\mathcal{R} = \{\mathcal{R}^1, \mathcal{R}^2, \dots, \mathcal{R}^n\}$  and a data collector  $\mathcal{C}$ . Each respondent  $i$  has a database  $\mathcal{D}_i$  with  $m$  records. We denote  $T$  as the dataset collected by the data collector. Also, the dataset  $T$  consists of  $d$  quasi-identifier  $\text{QID} = \{\text{QI}_1, \text{QI}_2, \dots, \text{QI}_d\}$  and a sensitive attribute. Note that the quasi-identifier can be either categorical or continuous data while the sensitive attribute is a categorical data from its domain (e.g., disease).

A quasi-identifier (QI) is a minimal set of attributes in  $T$  that can be joined with external information to uniquely distinguish individual records [24]. Note that the quasi-identifier can be either categorical or continuous data while the sensitive attribute is a categorical data from its domain.

*Definition 1* (quasi-identifier). A quasi-identifier (QI) is a minimal set of attributes that can uniquely distinguish tuples in  $T$ . The QI for Table 1 is  $\{\text{Gender}, \text{Age}, \text{Zip}\}$  and it can be generalized as  $\{\text{Male}, 10-16, 278 * *\}$ .

*Definition 2* ( $k$ -anonymity).  $T$  is said to satisfy  $k$ -anonymity with respect to QI if and only if each set of attributes in QI appears at least  $k$  occurrences in  $T$ .

*Definition 3* (self-awareness privacy). Each respondent  $i$  is said to achieve self-awareness privacy if he learns the protection level (e.g.,  $k$ -anonymity) provided by the data collector. At the end of the protocol execution, each respondent remains anonymous to others and the data collector is not able to identify any of the respondents with probability more than 0.5.

*4.3. Components.* Our self-awareness data collection protocol consists of the following three components.

- (i) *Data collector*: an authorized party who wants to collect data from a group of respondents via wired or wireless network.
- (ii) *Respondent*: participant in the data collection process who is also a candidate to submit his/her record to the data collector.
- (iii) *The onion router* (Tor): an anonymous network used to conceal the respondent's privacy such that the agency cannot monitor the activity flows of any respondent.

We show the interactions among the components in our solution in Figure 1. We assume that the respondents and the data collector are equipped with ubiquitous sensors to detect, communicate, and execute the protocol.

*4.4. Adversary Model.* We assume that both the data collector and the respondents are semihonest players (also known as honest-but-curious). Semihonest players follow the protocol faithfully but may try to discover extra information during the protocol execution.

In our protocol design, the data collector must follow the protocol faithfully in order to ensure that all respondents are willing to participate in the data collection process. For the same reason, all respondents should be semihonest in order to ensure that the privacy protection level offered by the data collector can be achieved.

*4.5. Notations Used.* The notations used hereafter in this paper are summarized in Notations section.

## 5. Self-Awareness Data Collection Protocol

*5.1. Protocol Idea.* The basic idea of our protocol is to allow the respondents to know the protection level they will receive from the data collector before the data submission process [35]. In our design, the data collector will release a set of quasi-identifiers  $\text{QID} = \{\text{QI}_1, \text{QI}_2, \dots, \text{QI}_n\}$  for  $T$  and define a protection level it wants to provide to the respondents (e.g., a threshold  $k$ ). Note that a larger  $k$  will make the respondents feel more comfortable to submit their records. We also require the respondents to collaborate together to find the number of records in  $(\mathcal{D}_1 \cup \mathcal{D}_2 \cup \dots \cup \mathcal{D}_n)$  which met the quasi-identifier determined by the data collector. We assume that the communication between the data collector and the respondents is via a mixture network such as Tor [14]. Note that the communication (respondents and data collector) and collaboration (among respondents) in our solution are run automatically. We show the overview of our proposed solution in Figure 1.

In the following sections, we will describe our self-awareness data collection protocol in details.

*5.2. Our Protocol.* In order to participate in the data collection process, all players can precompute some information to be used during the protocol execution. For example, each respondent  $i$  can generate a cryptographic key pair  $(\text{pk}_i, \text{pr}_i)$  where  $\text{pk}_i$  is the public key and  $\text{pr}_i$  is the corresponding private key. Next, the respondents encrypt their personal identifiable information (PII) such as name or social security number by using the  $\text{pk}_i$ . The encrypted PII will be used as the public identity  $\mathcal{I}_i$  of the respondent  $i$ . This public identity is important for other respondents to identify the owner of a given public key. Each respondent then submits his public identity and encryption key to the data collector via a Tor network. Let us assume there are  $n$  respondents who participate in the data collection process and, hence, the data collector will receive  $n$  submissions  $(\mathcal{I}_1, \text{pk}_1), (\mathcal{I}_2, \text{pk}_2), \dots, (\mathcal{I}_n, \text{pk}_n)$  from the respondents.

Before the data collection begins, the data collector is required to define a set of  $m$  quasi-identifiers denoted as  $\text{QID} = \{\text{QI}_1, \text{QI}_2, \dots, \text{QI}_m\}$  for the dataset  $T$  to be collected and determine the protection level (e.g.,  $k$  value) for the respondents.

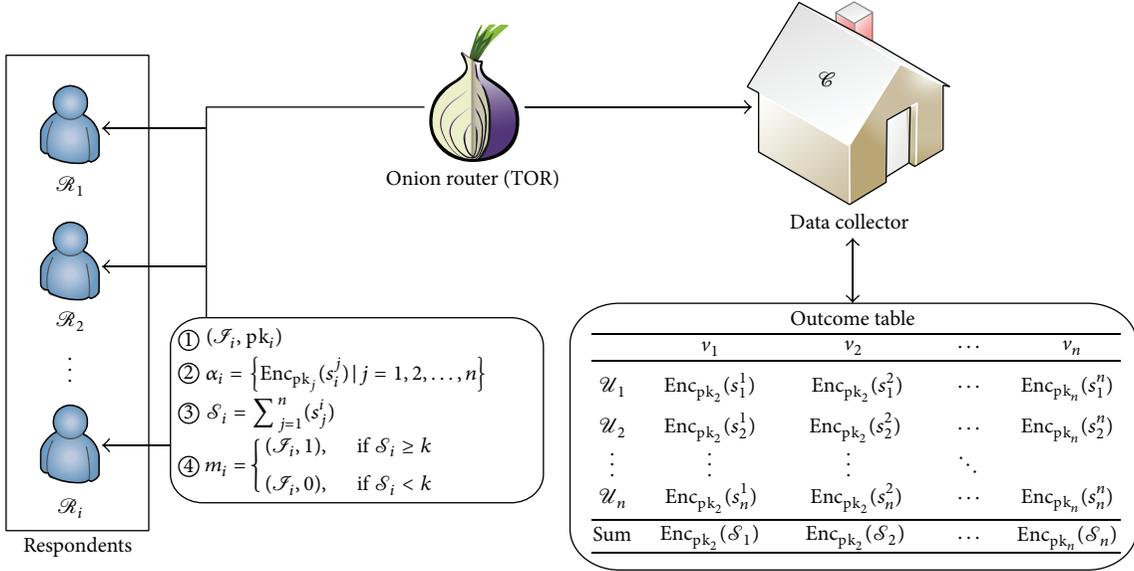


FIGURE 1: Overview of the proposed solution.

TABLE 2: Outcome table released by the data collector.

|          | $v_1$                             | $v_2$                             | $\dots$  | $v_n$                             |
|----------|-----------------------------------|-----------------------------------|----------|-----------------------------------|
| $u_1$    | $\text{Enc}_{\text{pk}_1}(s_1^1)$ | $\text{Enc}_{\text{pk}_2}(s_1^2)$ | $\dots$  | $\text{Enc}_{\text{pk}_n}(s_1^n)$ |
| $u_2$    | $\text{Enc}_{\text{pk}_1}(s_2^1)$ | $\text{Enc}_{\text{pk}_2}(s_2^2)$ | $\dots$  | $\text{Enc}_{\text{pk}_n}(s_2^n)$ |
| $\vdots$ | $\vdots$                          | $\vdots$                          | $\ddots$ | $\vdots$                          |
| $u_n$    | $\text{Enc}_{\text{pk}_1}(s_n^1)$ | $\text{Enc}_{\text{pk}_2}(s_n^2)$ | $\dots$  | $\text{Enc}_{\text{pk}_n}(s_n^n)$ |
| SUM      | $\text{Enc}_{\text{pk}_1}(S_1)$   | $\text{Enc}_{\text{pk}_2}(S_2)$   | $\dots$  | $\text{Enc}_{\text{pk}_n}(S_n)$   |

To initiate the protocol, the data collector first randomly assigns a public key  $\text{pk}_i$  for each  $\text{QI}_j \in \text{QID}$ . If  $|\text{QID}| > n$ , the same public key can be assigned to more than one quasi-identifier. Otherwise, the data collector selects  $m/n$  of the public keys for the assignment. For simplicity, we will assume that the size for both quasi-identifier and public key is equal (i.e.,  $m = n$ ) and  $\ell = \{(\text{pk}_1, \text{QI}_1), (\text{pk}_2, \text{QI}_2), \dots, (\text{pk}_n, \text{QI}_n)\}$ . Next, the data collector publishes  $(\mathcal{F}, \ell)$  to a shared location (e.g., a webpage):

$$(\mathcal{F}, \ell) = \{(\mathcal{F}_1, (\text{pk}_1, \text{QI}_1)), (\mathcal{F}_2, (\text{pk}_2, \text{QI}_2)), \dots, (\mathcal{F}_n, (\text{pk}_n, \text{QI}_n))\}. \quad (1)$$

Based on the information from (1), each respondent  $i$  retrieves  $\ell$  to examine if his records in  $\mathcal{D}_i$  match any of the quasi-identifiers  $\text{QI}_j \in \text{QID}$ . At this phase, each respondent  $i$  maintains a scores list for QID,  $\{s_i^1, s_i^2, \dots, s_i^n\}$ . We denote  $s_i^j$  as the score determined by the respondent  $i$  for  $\text{QI}_j$ . The respondent raises each score by 1 when a record in  $\mathcal{D}_i$  matches the quasi-identifier. Upon the completion, the respondent  $i$  encrypts each  $s_i^j$  by using the public key  $\text{pk}_j$  assigned to the quasi-identifier  $\text{QI}_j$ . The encrypted scores list computed by each respondent  $i$  can be represented as  $\alpha_i =$

$\{\text{Enc}_{\text{pk}_1}(s_i^1), \text{Enc}_{\text{pk}_2}(s_i^2), \dots, \text{Enc}_{\text{pk}_n}(s_i^n)\}$ . Then, all the respondents send  $\alpha_i$  to the data collector and a shared location. Note that this location can be a separate space that is not shared with the data collector.

Upon receiving  $\alpha_i$  from all the respondents, the data collector performs the following tasks.

- (1) *Aggregates the scores determined by all respondents for each  $\text{QI}_j$ .* The data collector performs this computation in an encrypted form by using the additive property of the Paillier cryptosystem. The output of the aggregation can be represented as

$$\begin{aligned} \text{Enc}_{\text{pk}_j}(S_j) &= \text{Enc}_{\text{pk}_j}(s_1^j) +_n \text{Enc}_{\text{pk}_j}(s_2^j) \\ &+_n \dots +_n \text{Enc}_{\text{pk}_j}(s_n^j). \end{aligned} \quad (2)$$

- (2) *Publishes an outcome table.* The data collector publishes the scores for each  $\text{QI}_j$  in an outcome table as shown in Table 2. In Table 2, each row ( $u_i$ ) represents the encrypted scores received from each respondent  $i$  while the column ( $v_j$ ) shows the encrypted scores for each quasi-identifier  $\text{QI}_j$ . Note that all the data in  $v_j$  are encrypted by using the same public key  $\text{pk}_j$ . Therefore, only the respondent who has been assigned the  $\text{QI}_j$  can decrypt  $\text{Enc}_{\text{pk}_j}(S_j)$  to learn the number of matched records ( $S_j$ ) for  $\text{QI}_j$ .

After the data collector releases the outcome table, the respondents need to verify that the data released are genuine. For instance, each respondent  $i$  verifies that the encrypted scores list  $\alpha_i$  submitted to the data collector appears as one of the rows in Table 2. If the respondent fails to verify the data, he or she then issues a decision message  $m_i$  with a random value.

Let us assume all the respondents successfully verify the data in Table 2. Next, each respondent  $i$  retrieves  $v_j$

**Self-Awareness Data Collection Protocol****Phase 1: Public Key and Public Identity Submissions**

The data collector broadcasts a submission request to  $n$  respondents. Each  $\mathcal{R}_i$  generates a cryptographic key pair  $(pk_i, pr_i)$  and a public identity  $\mathcal{F}_i$  by encrypting its personal identifiable information (PII). Note that the respondents can pre-compute the cryptographic key pair and the PII in an offline mode. Next, each  $\mathcal{R}_i$  sends  $(\mathcal{F}_i, pk_i)$  to  $\mathcal{C}$  via the Tor network.

**Phase 2: Satisfaction Scores Computation**

The data collector  $\mathcal{C}$  generates QID, decides a threshold  $k$  and assigns a public key for each  $QI_i$ . Next, it broadcasts the information to all respondents. Each  $\mathcal{R}_i$  examines if his record in  $\mathcal{D}_i$  satisfy QID. For each satisfy case, the  $\mathcal{R}_i$  increases the constraint score  $s_i^j$  by 1. We denote  $s_i^j$  as the score determines by  $\mathcal{R}_i$  for  $QI_j$ . Next, each  $\mathcal{R}_i$  encrypts  $\{s_i^j \mid j = 1, 2, \dots, n\}$  by using the public key  $pk_j$  to produce  $\alpha_i = \{\text{Enc}_{pk_j}(s_i^j) \mid j = 1, 2, \dots, n\}$ . Each  $\mathcal{R}_i$  then anonymously sends  $\alpha_i$  to  $\mathcal{C}$  and a shared location.

**Phase 3: Scores List Verification**

The data collector  $\mathcal{C}$  computes and publishes an outcome table. Each  $\mathcal{R}_i$  examines if the published scores list is same as the original list he sent to  $\mathcal{C}$ . If the list has been modified, the respondent will not participate in the next phase.

**Phase 4: Satisfaction Score Checking**

Each  $\mathcal{R}_j$  retrieves and decrypts  $\{\text{Enc}_{pk_j}(s_i^j) \mid i = 1, 2, \dots, n\}$ . Next, it computes  $\mathcal{S}_j = \sum_{i=1}^n (s_i^j)$  as the satisfaction score for  $QI_j$ . If the satisfaction score  $\mathcal{S}_j$  is at least with  $k_j$  occurrences (e.g.,  $\mathcal{S}_j \geq k_j$ ), the  $\mathcal{R}_j$  sends  $m_i = (\mathcal{F}_i, 1)$  to  $\mathcal{C}$ . Otherwise,  $m_i = (\mathcal{F}_i, 0)$  will be sent to  $\mathcal{C}$ .

**Phase 5: Data Submission**

The respondents submit his record to  $\mathcal{C}$  with the confidence that their privacy protection is achieved at  $k$ -anonymity level.

ALGORITHM 1: Self-Awareness data collection protocol.

(based on his public identity  $\mathcal{F}_i$ ) and decrypts all encrypted data by using the private key  $pr_i$ . After the decryption, the respondents must ensure that the aggregated score  $\text{Enc}_{pk_j}(\mathcal{S}_i)$  computed by the data collector is correct. The respondents can verify this by computing  $\mathcal{S}_i = \sum_{j=1}^n (s_j^i)$  from the decrypted scores and then compare it with the decrypted result of  $\text{Enc}_{pk_j}(\mathcal{S}_i)$ . Lastly, each respondent  $i$  compares  $\mathcal{S}_i$  with the threshold  $k$  determined by the data collector. If the number of matched records  $\mathcal{S}_i$  is greater than the threshold value (e.g.,  $\mathcal{S}_i \geq k$ ), we assume that the respondent will submit his records to the data collector. Otherwise, the respondent will abort from the data collection process.

At the final phase, each respondent  $i$  sends a decision message  $m_i$  to the shared location. If the decision message  $m_i$  is set to 1, this indicates that  $\mathcal{S}_i \geq k$ . Therefore, the respondents should submit their records to the data collector. Otherwise, if  $m_i$  is set to 0, the respondents should not reveal any record to the data collector.

We summarize our self-awareness data collection protocol in Algorithm 1.

**6. Analysis and Discussion**

**6.1. Analysis of Correctness.** In this paper, we assume that both the data collector and the respondents are semihonest players. The semihonest model is realistic in our solution. If

both players follow the protocol faithfully, each respondent can ensure that he will achieve the protection level offered by the data collector (e.g.,  $k$ -anonymity). At the same time, the data collector can guarantee that the datasets collected are useful for analysis.

During the protocol execution, all respondents are required to verify (1) the encrypted scores released by the data collector are genuine and (2) the aggregated score for each  $QI_j$  computed by the data collector is correct. The first verification is to ensure that the data collector has received all data computed by the respondents correctly while the second verification is useful for the respondents to detect a malicious data collector.

In our protocol design, the data collector needs to define a protection level (e.g.,  $k$  value) before the data collection begins. The data collector can define the same protection level for all  $QI_j$  or define difference in anonymous levels  $k_i$  for each  $QI_i \in QID$ . For the latter case, the respondents can perform the same steps to verify each value of  $k_i$ .

**6.2. Analysis of Privacy.** The privacy analysis of our protocol depends on how much information has been revealed during the protocol execution. In general, our solution should protect the privacy of the respondents. This leads to the following two requirements: (1) the data collector should not be able to infer any sensitive information of the respondents from the data collected and (2) the respondents are aware of

the data they submit and the protection level they will receive from the data collector.

In our protocol design, we utilize Tor network to prevent direct communication between the data collector and the respondents. This approach will not allow the data collector to track the identity of any respondent. Also, we assume that each respondent has no knowledge about the profile of other respondents, but the number of respondents in the protocol is known publicly.

The unique identity  $\mathcal{S}_i$  of each respondent will not leak the profile of any respondent because they are in an encrypted form. The data collector is not able to decrypt  $\alpha_i$  in the absence of private keys from the respondents. Further, our protocol ensures that no party (including the data collector) can learn the encrypted score in the outcome table before the decryption. Note that only the respondent who has the private key can perform the decryption.

To prevent possible collusions between the data collector and other respondents, we assume that all data transmissions are performed via an anonymous communication channel (e.g., Tor network). This can ensure that the profile of each respondent remains anonymous from others.

The shared location (e.g., web page or web folder) used in our protocol is to allow the respondents to learn the decisions made by others and to detect a malicious data collector. Each respondent notifies others about the verification result by using a decision message  $m$ . Since the decision message only reveals the public identity of the respondents, we can assume that the profile of the respondents remains hidden from others.

**6.3. Analysis of Efficiency.** The complexity of our protocol is dominated by the cryptographic operations (encryption and decryption) performed by respondents. We implement our protocol in Java and ran it on a single computer with a 2 GHz CPU and a 2 GB RAM. The performance evaluation is shown in Figure 2. Each respondent performs the same amount of cryptographic operations in our experiment.

**6.4. Discussion.** In this paper, we assume that the size of the public keys (or the number of respondents) and the quasi-identifier is equal (e.g.,  $|\mathcal{R}| = |\text{QID}| = n$ ). However, our protocol works correctly for unequal cases. The owner of the public key only performs the decryption and computes  $\mathcal{S}_i$  at the end of the protocol execution. A respondent may not be involved in the final phase if his public key is not selected by the data collector (for cases when  $|\mathcal{R}| > n$ ). Otherwise, a respondent needs to repeat final phase for several times if his public key is assigned to more than one  $\text{QI}_j$ .

## 7. Conclusion and Future Work

In this paper, we presented a self-awareness protocol for IoT data collection. Since the release of raw data to the data collector has a high risk to compromise privacy of the respondents, we aim to increase confidence of the respondents before they submit their records to the data collector. Our self-awareness protocol allows each respondent to help others in

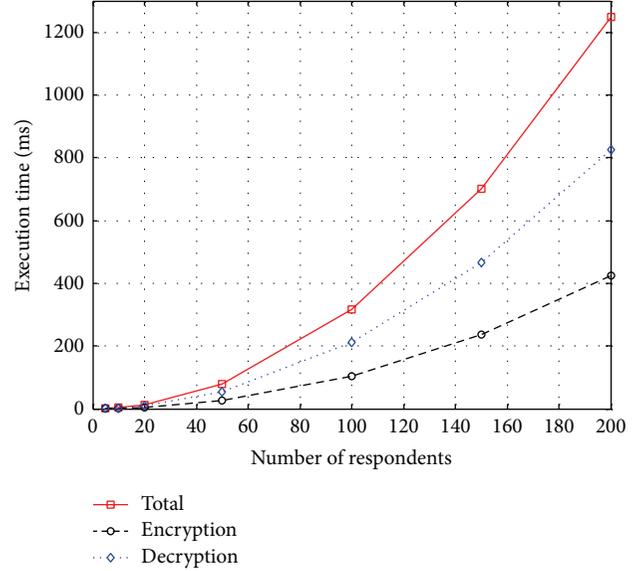


FIGURE 2: Performance of the proposed solution.

order to preserve his own privacy. At the same time, the final collected data should adhere to the protection level promised by the data collector before the data collection begins. Also, our solution can be extended to support indictment scheme (when the data is released to a third party) because the respondents have evidence (e.g., value of  $k$ ) to indict a malicious data collector.

## Notations

|                                     |                                                          |
|-------------------------------------|----------------------------------------------------------|
| $\mathcal{R}_i$ :                   | Respondent $i$                                           |
| $ \mathcal{R} $ :                   | Size of the respondents                                  |
| $T$ :                               | Dataset collected by the data collector                  |
| $\mathcal{D}_i$ :                   | Local database of respondent $i$                         |
| $k$ :                               | Anonymous protection level                               |
| QID:                                | Quasi-identifier set determined by the data collector    |
| $ \text{QID} $ :                    | Size of the quasi-identifier                             |
| $\text{QI}_i$ :                     | $i$ th quasi-identifier in QID                           |
| $\mathcal{S}_i$ :                   | Public identity of the respondent $i$                    |
| $s_i^j$ :                           | Score determined by the respondent $i$ for $\text{QI}_j$ |
| $\mathcal{S}_i$ :                   | Satisfaction score of $\text{QI}_i$                      |
| $\text{pk}_i$ :                     | Public key of respondent $i$                             |
| $\text{pr}_i$ :                     | Private key of respondent $i$                            |
| $\text{Enc}_{\text{pk}_i}(\cdot)$ : | Encryption operation by using $\text{pk}_i$              |
| $\text{Dec}_{\text{pr}_i}(\cdot)$ : | Decryption operation by using $\text{pr}_i$              |
| $m_i$ :                             | Decision message from respondent $i$ .                   |

## Conflict of Interests

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

## References

- [1] N. Y. Yen and S. Y. F. Kuo, "An integrated approach for internet resources mining and searching," *Journal of Convergence*, vol. 3, pp. 37–44, 2012.
- [2] S. K. Dhurandher, M. S. Obaidat, and M. Gupta, "An acoustic communication based AQUA-GLOMO simulator for underwater networks," *Human-Centric Computation and Information Sciences*, vol. 2, article 3, 2012.
- [3] B. Singh and D. K. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computation and Information Sciences*, vol. 2, article 13, 2012.
- [4] G. H. S. Carvalho, I. Woungang, A. Anpalagan, and S. K. Dhurandher, "Energy-efficient radio resource management scheme for heterogeneous wireless networks: a queueing theory perspective," *Journal of Convergence*, vol. 3, no. 4, pp. 15–22, 2012.
- [5] A. U. Bandaranayake, V. Pandit, and D. P. Agrawal, "Indoor link quality comparison of IEEE 802.11a channels in a multi-radio mesh network testbed," *Journal of Information Processing Systems*, vol. 8, no. 1, pp. 1–20, 2012.
- [6] J. K.-Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.
- [7] F. Barigou, B. Atmani, and B. Beldjilali, "Using a cellular automaton to extract medical information from clinical reports," *Journal of Information Processing Systems*, vol. 8, no. 1, pp. 67–84, 2012.
- [8] R. H. Weber, "Internet of things—new security and privacy challenges," *Computer Law and Security Review*, vol. 26, no. 1, pp. 23–30, 2010.
- [9] N. Zhong, J. H. Ma, R. H. Huang et al., "Research challenges and perspectives on wisdom web of things (W2T)," *The Journal of Supercomputing*, vol. 64, no. 3, pp. 862–882, 2013.
- [10] J. Ma, "Smart u-things-challenging real world complexity," *IPSI Symposium Series*, vol. 19, pp. 146–150, 2005.
- [11] J. Ma, "Smart u-things and ubiquitous intelligence," in *Proceedings of the 2nd International Conference on Embedded Software and Systems*, p. 776, Springer, Xi'an, China, 2005.
- [12] J. Ma, Q. Zhao, V. Chaudhary et al., "Ubisafe computing: vision and challenges (I)," in *Proceedings of the 3rd International Conference on Autonomic and Trusted Computing*, pp. 386–397, Springer, Wuhan, China, 2006.
- [13] J. Ma, L. T. Yang, B. O. Apduhan, R. Huang, L. Barolli, and M. Takizawa, "Towards a smart world and ubiquitous intelligence: a walkthrough from smart things to smart hyperspaces and UbiKids," *International Journal of Pervasive Computing and Communications*, vol. 1, no. 1, pp. 53–68, 2005.
- [14] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: the second-generation onion router," in *Proceedings of the 13th conference on USENIX Security Symposium*, vol. 13, p. 21, USENIX Association, San Diego, Calif, USA, 2004.
- [15] M. Edman and B. Yener, "On anonymity in an electronic society: a survey of anonymous communication systems," *ACM Computing Surveys*, vol. 42, no. 1, article 5, 2009.
- [16] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–88, 1981.
- [17] K. Peng, "Attack and correction: how to design a secure and efficient mix network," *Journal of Information Processing Systems*, vol. 8, no. 1, pp. 175–190, 2012.
- [18] B. Li, E. Erdin, M. H. Gunes, G. Bebis, and T. Shipley, "An analysis of anonymity technology usage," in *Proceedings of the 3rd International Conference on Traffic Monitoring and Analysis*, pp. 108–121, Springer, Vienna, Austria, 2011.
- [19] J. Domingo-Ferrer, "Copriacy: towards a theory of sustainable privacy," in *Proceedings of the International Conference on Privacy in Statistical Databases*, pp. 258–268, Springer, Corfu, Greece, 2010.
- [20] J. Domingo-Ferrer, "Copriacy: an introduction to the theory and applications of co-operative privacy,"  *SORT: Statistics and Operations Research Transactions*, pp. 25–40, 2011.
- [21] T. Teraoka, "Organization and exploration of heterogeneous personal data collected in daily life," *Human-Centric Computing and Information Sciences*, vol. 2, article 1, 2012.
- [22] Privacy Online: A Report to Congress. Federal Trade Commission, 1998.
- [23] K.-S. Wong and M. Kim, "Secure re-publication of dynamic big data," in *Cyberspace Safety and Security*, G. Wang, I. Ray, D. Feng, and M. Rajarajan, Eds., vol. 8300, pp. 468–477, Springer International Publishing, 2013.
- [24] L. Sweeney, "k-anonymity: a model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.
- [25] P. Samarati and L. Sweeney, "Generalizing data to provide anonymity when disclosing information (abstract)," in *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, p. 188, ACM, Seattle, Wash, USA, 1998.
- [26] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkatasubramanian, "l-diversity: privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, article 3, 2007.
- [27] R. C.-W. Wong, J. Li, A. W.-C. Fu, and K. Wang, "(alpha, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 754–759, ACM, Philadelphia, Pa, USA, 2006.
- [28] R. C.-W. Wong, Y. Liu, J. Yin, Z. Huang, A. W.-C. Fu, and J. Pei, "(alpha, k)-anonymity based privacy preservation by lossy join," in *Proceedings of the Joint 9th Asia-Pacific Web and 8th International Conference on Web-Age Information Management Conference on Advances in Data and Web Management*, pp. 733–744, Springer, Huang Shan, China, 2007.
- [29] L. Ninghui, L. Tiancheng, and S. Venkatasubramanian, "t-Closeness: privacy beyond k-anonymity and l-diversity," in *Proceedings of the 23rd International Conference on Data Engineering (ICDE '07)*, pp. 106–115, Istanbul, Turkey, April 2007.
- [30] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: a survey of recent developments," *ACM Computing Surveys*, vol. 42, no. 4, article 14, 2010.
- [31] P. Golle, F. McSherry, and I. Mironov, "Data collection with self-enforcing privacy," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06)*, pp. 69–78, ACM, Alexandria, Va, USA, November 2006.
- [32] M. Stegelmann, "Towards fair indictment for data collection with self-enforcing privacy," in *Security and Privacy—Silver Linings in the Cloud*, K. Rannenberg, V. Varadharajan, and C. Weber, Eds., vol. 330, pp. 265–276, Springer, Berlin, Germany, 2010.
- [33] R. Kumar, R. Gopal, and R. Garfinkel, "Freedom of privacy: anonymous data collection with respondent-defined privacy

protection,” *INFORMS Journal on Computing*, vol. 22, no. 3, pp. 471–481, 2010.

- [34] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proceedings of the 17th International Conference on Theory and Application of Cryptographic Techniques*, pp. 223–238, Springer, Prague, Czech Republic, 1999.
- [35] K.-S. Wong and M. Kim, “Privacy-preserving data collection with self-awareness protection,” in *Frontier and Innovation in Future Computing and Communications*, J. J. Park, A. Zomaya, H.-Y. Jeong, and M. Obaidat, Eds., vol. 301, pp. 365–371, Springer, Amsterdam, The Netherlands, 2014.

## Research Article

# A Study on Intelligent User-Centric Logistics Service Model Using Ontology

**Saraswathi Sivamani, Kyunghun Kwak, and Yongyun Cho**

*Information and Communication Engineering, Suncheon National University, 413 Jungangno, Suncheon, Jeonnam 540-742, Republic of Korea*

Correspondence should be addressed to Yongyun Cho; [yycho@sunchon.ac.kr](mailto:yycho@sunchon.ac.kr)

Received 30 January 2014; Accepted 6 April 2014; Published 19 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Saraswathi Sivamani 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.

Much research has been undergone in the smart logistics environment for the prompt delivery of the product in the right place at the right time. Most of the services were based on time management, routing technique, and location based services. The services in the recent logistics environment aim for situation based logistics service centered around the user by utilizing various information technologies such as mobile devices, computer systems, and GPS. This paper proposes a smart logistics service model for providing user-centric intelligent logistics service by utilizing smartphones in a smart environment. We also develop an OWL based ontology model for the smart logistics for the better understanding among the context information. In addition to basic delivery information, the proposed service model makes use of the location and situation information of the delivery vehicle and user, to draw the route information according to the user's requirement. With the increase of internet usage, the real-time situations are received which helps to create a more reliable relationship, owing to the Internet of Things. Through this service model, it is possible to engage in the development of various IT and logistics convergence services based on situation information between the deliverer and user which occurs in real time.

## 1. Introduction

Logistics services have improved immensely, after the emergence of ubiquitous computing [1]. In the early days, the logistics services experienced more time related constraints due to the traffic and frequent change of the customer environment. Many researches have been underway to reduce the time and effort which can help in providing a faster, safer, and more accurate delivery service. Such efforts have been put forth in the practical development by taking advantage of various advanced computing technologies and smart communication devices in the logistics service environment which is well explained by Klaus [2].

Most popular research includes the optimal routing technique [3] and the logistic tracking by using wireless telecommunication technology based on RFID or GPS [4] for more reliable service. Many of such services include the user interaction. The smart computing environment centered on individual smartphone users, which is recently growing rapidly [5], is demanding the evolution of human-centered

computing services that can reflect the user's opinion and surrounding situation more directly in various service areas. In other words, the immediate action on the situational context in the logistics process according to the user's requirement and the frequent change in the environment can improve the service quality. All the logistic services are focused on the user which drives to build a fast reliable service [6]. For the situation based decision making, we need a shared understanding among the context information to gain complete awareness [7]. Using ontology in the service model enables the understanding of the relationship between the context and situation information.

This paper proposes a model based on the user-centric smart logistics/delivery services considering various situational information which arise in ubiquitous computing environment. In addition, the context based ontology model for the proposed model is developed to deal with the semantic situations. The proposed service model makes use of GPS-based tracking technology as well as various types of situation information based on sensing information technologies from

various sensors in a smart space as limiting factors of the route and schedule planning of logistics/delivery services. Users are allowed to track, interact, and change the location through the GUI-based application developed for the smart phone. For this, the proposed service model includes GUI-based situation information processing technology using smartphones and context-aware GUI-based logistics service scenario and model technologies which use smartphones.

The rest of the paper is organized as follows. In the Section 2, the related studies for the designed model are designed. Next, Section 3 discusses the situation in logistics with the context information. Section 4 discusses the service model of the logistics process. Scenario based route decision and logical view of the logistics service is explained in Section 5 and the final section provides the conclusion and future work.

## 2. Related Studies

In recent times, there are active studies underway on logistics automation technologies and IT based logistics smart service technologies in the ubiquitous environment. In particular, practical technology researches [8] are underway on smartphones and RFID/USN technologies for providing context-aware services in various areas from shopping malls to logistics, home delivery, and ship and air freight. It proves that the environment is entering a new dimension with the Internet of Things [9]. Scholz-Reiter [10] explains the automation process in logistics with the RFID transponder that locates the objects on a real-time. Related studies on logistics environment, using location data and RFID have proposed the convergence of IT technologies and logistics regarding various application areas such as control of delivery vehicles using location data based on RFID, method for logistics information and event notification, and management of logistics warehouse [11–14].

Such research has great significance in the fact that it enhances the efficiency of logistics and delivery work by grafting computing models and networking technologies into traditional logistics related work processes. Another impact factor in the logistic management is the vehicle routing problem (VRP). Many technologies and algorithms were developed to acquire an optimal routing of the transportation. Peng et al. [15] attempt to resolve the VRP problem through the hybrid computational intelligence where some approaches include various algorithms to find the optimum route [16, 17]. In addition, such studies have developed into studies which attempt to consider the simple data values which are sensed from a sensor in a logistics system and process management and control, along with various types of situation information such as user and environmental information, as the flow elements of the logistics/delivery process. Through these efforts, several researchers have defined situation information for logistics/delivery environment and have proposed methods and models for recognizing and processing situation information for logistics/delivery services and systems [18–20].

Usage of the phone has improved the productivity in various consumer-producer [21] and receiver-deliverer relationships for safe transactions. A case study on logistics/delivery services using smartphones presented its usefulness by examining various events which occur when employees in a USA logistics/delivery company were actually prepared to use a logistics/delivery system using smartphones [22]. According to research findings, Internet based logistics/delivery control system using smartphones was introduced rather than the traditional PDA-based logistics/delivery computer system [23]. As many studies proved that the user-centric service has enhanced the productivity [24], it also had a positive effect on the user convenience of people in charge of logistics/delivery using the devices.

Presently, it is anticipated that the services using smartphones will have a very wide range and diversity of applicable service domains [5] and will also benefit many applications [25]. However, most studies [26–28] are being concentrated in the tracking of logistics/delivery based on simple position information using GPS and optimization of routes, and there is a lack of studies on the models that can apply various types of sensing information communicated from RFID/USN of the logistics/delivery environment. Thus, there is a need for the studies on smart logistics/delivery service model that can provide the optimum logistics/delivery service according to user-based situation information arising in real time in the actual logistics delivery environment.

A study of a controlling system for the logistics vehicle using a smartphone proposed a design, which overcomes the limitations of a logistics vehicle control system using only the existing GPS-based location information and for a smooth logistics vehicle control system in a limited time and location using the 3G network of smartphones [29]. Along with the vehicle tracking and routing technique, the situation aware system is required for a user centered service in the logistic system. Howard and Cambria [30] explains the importance of the situation awareness in the human centric environment where the situation is subjected to change in unexpected variations. The perspective of the situation varies for each service in which Yau et al. [31] defines situations as well as contexts and classifies situations into atomic and composite ones.

The situation aware context information needs to be processed in the automated logistic service to generate an optimal route according to the customer's requirement. For this purpose, the OWL based ontology is used which is the ontology representation language. Many studies are underway in the ontology based logistics services to enhance the productivity and time efficiency in the delivery.

## 3. Situation in Logistics Process

In the dynamic computing environment, situation awareness is more important for decision making system. As our proposed system revolves around the user change, the situational change takes its turn. The set of context information obtained determines the situation of logistic and provides optimal route information according to the customer's needs.

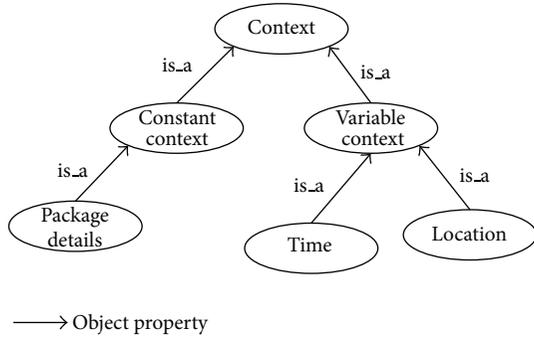


FIGURE 1: Ontology tree of context class.

To provide a user based service, the context information is required, which can be classified into constant context and variable context. The constant context consists of unchangeable information, whereas the variable context contains the changeable information that may create a situation in the logistic process. The package details of the customer are certain and fixed information that is linked with the corresponding customer. The variable contexts in the logistic are time and location that are related to both the deliverer and the receiver. Figure 1 presents the ontology tree of the context class.

The location of the vehicle and the customer are the basic changes in the situation logistics. The context information about the location is obtained through the communication network. The location concept in the ontology is shown in Figure 2. The vehicle's location is obtained with the GPS tracking system with the coordinates X and Y. Customer will be able to view the location of the vehicle from their smartphone to keep track of the product. Logistics App is installed on the customer's smartphone which not only helps in the tracking but also keeps discovering the location of the customer by activating the discover button. The customer's location is originally the address provided for the logistic, but the customer is permitted to modify the location when the vehicle crosses their nearby location. Also, by turning on the discovery tab, the location of the customer is made visible in the deliverer's smart device. According to the situation of the customer, the location and time can be changed and fulfill the customer needs.

#### 4. Service Model for an Intelligent Logistic Environment

Generally speaking, the actual smart logistics/delivery environment exists in the ubiquitous or intelligent space where various sensors connected to USN carry out mutual computing through networking with wired/wireless communication network. The general structure chart of the service model proposed by this paper for providing a smartphone-based smart logistics/delivery services linked to various sensors in a ubiquitous computing environment is as depicted in Figure 3.

The proposed smart logistics/delivery service model consists of the server part of providing the smart logistics/delivery service information based on the actual sensed

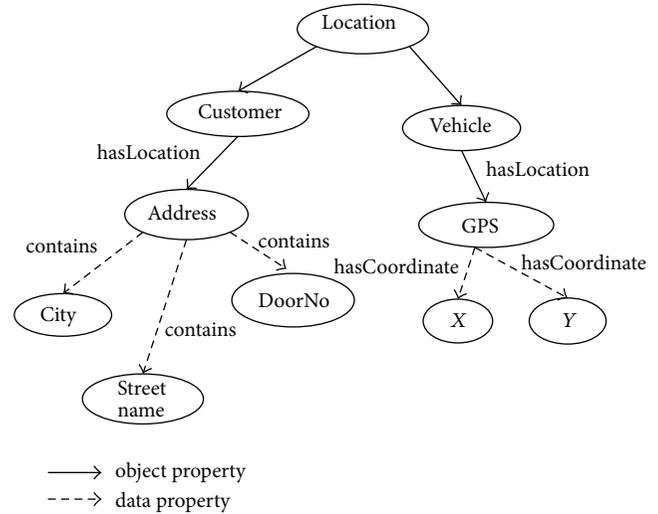


FIGURE 2: Location concept of the logistic ontology.

situation information to the person in charge of logistics/delivery and the user. The situation information from the user's smartphone is attained in the server of the smart logistics service to determine the route information about the logistics. The route of the delivery is determined by the user's location and availability. As noted by Gruber [32], ontology is a formal, explicit specification of a shared conceptualization. So, by taking advantage of the concept, understanding between the system and situation benefits the logistics service by providing a precise delivery.

4.1. High Context View of the Smart Logistics. The server receives the input of situation information, including the location of the sensed or entered logistics/delivery vehicle, location of the customer, status of the goods, time and method of how the customer received the delivered goods, and the real-time route change information of the person in charge of logistics/delivery.

In this case, the context model based on the rule of Figure 3 objectifies the low level terminal status information from the client through the RDF-based context model in the form of the status information class form of API level. Objectified low level terminal status information can be processed into high level status information with a richer meaning through the repetitive rule coupling with other objectified low level terminal status information. Figure 4 shows the conceptual diagram of the logistic process of the proposed service model for generating high level status information through coupling which has applied rules through the coupling applied with rules. The low level data delivered from various sensors is constituted into types and values according to the ontology knowledge dictionary related to the stored logistics/delivery domain in the ontology storage. A low level data represented by types and values is objectified into higher level status information through the repetitive coupling process with other low level data. For example, the coordinate information  $\langle X : Y \rangle$  representing the user location transmitted from a position sensor and GPS in

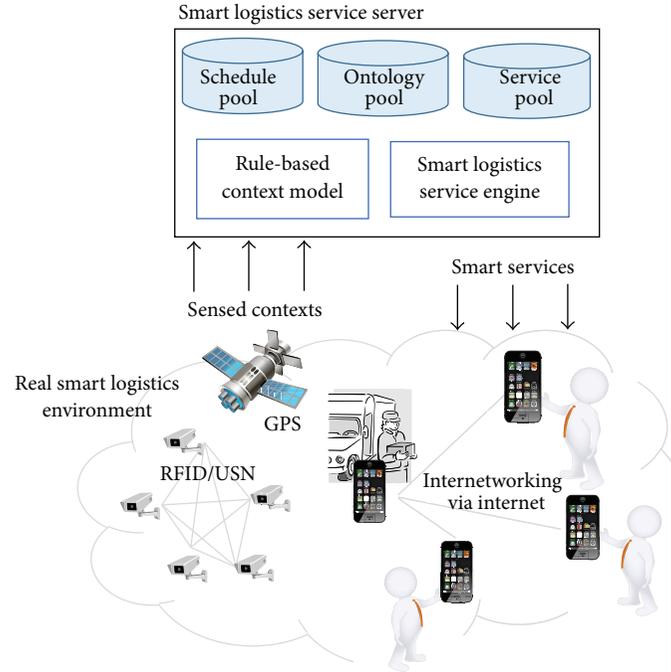


FIGURE 3: Service model of logistics service.

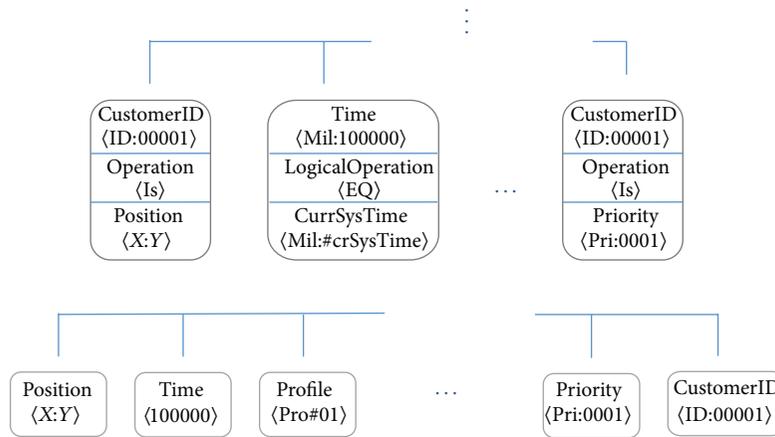


FIGURE 4: High level context information.

Figure 4 is expressed as a pair of position type for showing the actual integer and the semantic information of that value. The generated {Position,  $\langle X : Y \rangle$ } information couples with {CustomerID,  $\langle ID : 00001 \rangle$ } information for recognizing a specific user, and the user position with ID:00001 can be abstracted into a higher level information called  $\langle X : Y \rangle$ .

The smart logistics/delivery service provides various logistics/delivery related services ranging from the service for providing logistics/delivery route optimization information to the logistics/delivery automatic control service in the form of web service from the service storage using objectified high level status information. Then, the service engine uses the logistics/delivery plan information and the high level status information stored in the schedule storage as a limiting

factor for executing a specific service. The client sends the sensing information occurring in real-time from various sensors and the RFID and GPS devices installed on the logistics/delivery vehicles to the server. Then, the sensed information can be networked with the user’s smart device, and the user and the person in charge of logistics/delivery can be communicated through their smart devices. Through this kind of networking, the immediate user requirements can be considered in real time for the new services.

### 5. Logistics Ontology

User-Centric Intelligent Logistics Service uses OWL based ontology approach for fulfilling the customer satisfaction.

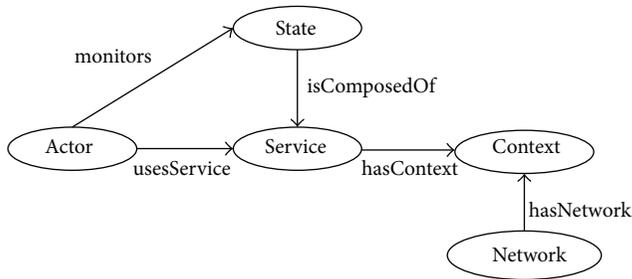


FIGURE 5: Top level ontology model of logistics service.

Usually the ontology contains the concepts, relations, axioms, and individuals. As a first step in the ontology development, the basic concepts of the logistics centered around the user are identified. The concepts are classified as network, state, service, context, and actor concepts. Figure 5 shows the Upper Ontology of the Logistics. The Concepts are briefly explained in the following subsections.

**5.1. Context Based Concept.** The context information is obtained from the smart space through the proper means of smart devices. The concept context in the smart logistics consists of time, location, and the information on the product delivered. As mentioned in the previous section, the context can be briefly classified into constant and variable context considering both the deliverer and customer.

**5.2. Network Based Concept.** The location information of the deliverer and the customer are considered of equal importance in the delivery of the product. Therefore, network is kept in check for the constant update of information. Location of the deliverer is obtained using GPS connection. The delivery time is calculated and sent to the customer with the location of the vehicle.

**5.3. State Based Concept.** According to the event and the action performed, the state of the delivery process is updated. Each stage of the process is updated on the server and presented to the customer. When the state of the process is in “On Delivery” state, the customer will be able to track down the delivery vehicle and also update personal change in the schedule. But once the delivery is done and state is converted to “Delivered” state, the customer will end all their connections from the logistics.

**5.4. Service Based Concept.** User-centric service is the main focus of the paper. The service revolves around the routing of the delivery with the situation based context information.

**5.5. Actor Based Concept.** In our model, the role of the actor is determined to be a customer or a driver. The basic information of the receiver and the sender is stored in the actor concept. In addition to this, the logistics company can also be added as one of the actors in broader vision.

The ontology is modelled using the protégé with OWL plugin. The class, object property, data property, and individual assertions are all created for a prototype modelling. Figure 6 shows the logistics ontology which explains the class relationship among the derived concepts.

## 6. User-Centric Scenario for Smart Logistics

Let us discuss a prototype scenario of the smart logistics with the situation and events that surround the customer and the deliverer.

- (i) Customers A, B, and C are planning to receive their delivered goods at each of their specific locations.
- (ii) The person in charge of the field Y of smart logistics/delivery company X initially receives the delivery route in the order of A, B, and C.
- (iii) Customer A recognizes that they currently have a personal matter to attend to and communicates this to the delivery company using their smart device.
- (iv) Customer B was to personally receive the delivered goods at their home, but B decided to briefly go out nearby.
- (v) Customer C was to personally receive the delivered goods at his home, but did not arrive at his house yet.

Now, we have some possible situations arising in the above mentioned scenario. With the received context information, a new route adjustment is derived and sent to the deliverer by exercising the proposed service model.

**6.1. Route Management Based on User Situation.** For each possible situation change, the route information is altered and sent to the deliverer. Considering the above mentioned scenario, the possible route changes are discussed as follows.

- (1) The time information from the private schedule entered by Customer A is objectified into new status information, and the existing logistics/delivery route is changed using the limitation information for setting a new route. Then, the newly determined route changes recommendation information as indicated on the smart device of the person in charge, and the person in charge continues the logistics/delivery service through the newly changed route (B → C → A).
- (2) While Customer B is briefly moving near their home where they were initially supposed to receive the goods, they directly send their own location information to the server using their smart device. Or the location of Customer B is automatically sent to the server through the sensor linked via network to a smart space and the networking with Customer B’s smart device.
- (3) New status information is objectified from Customer B’s location information and used as the limiting information for setting a new route. Then, if Customer

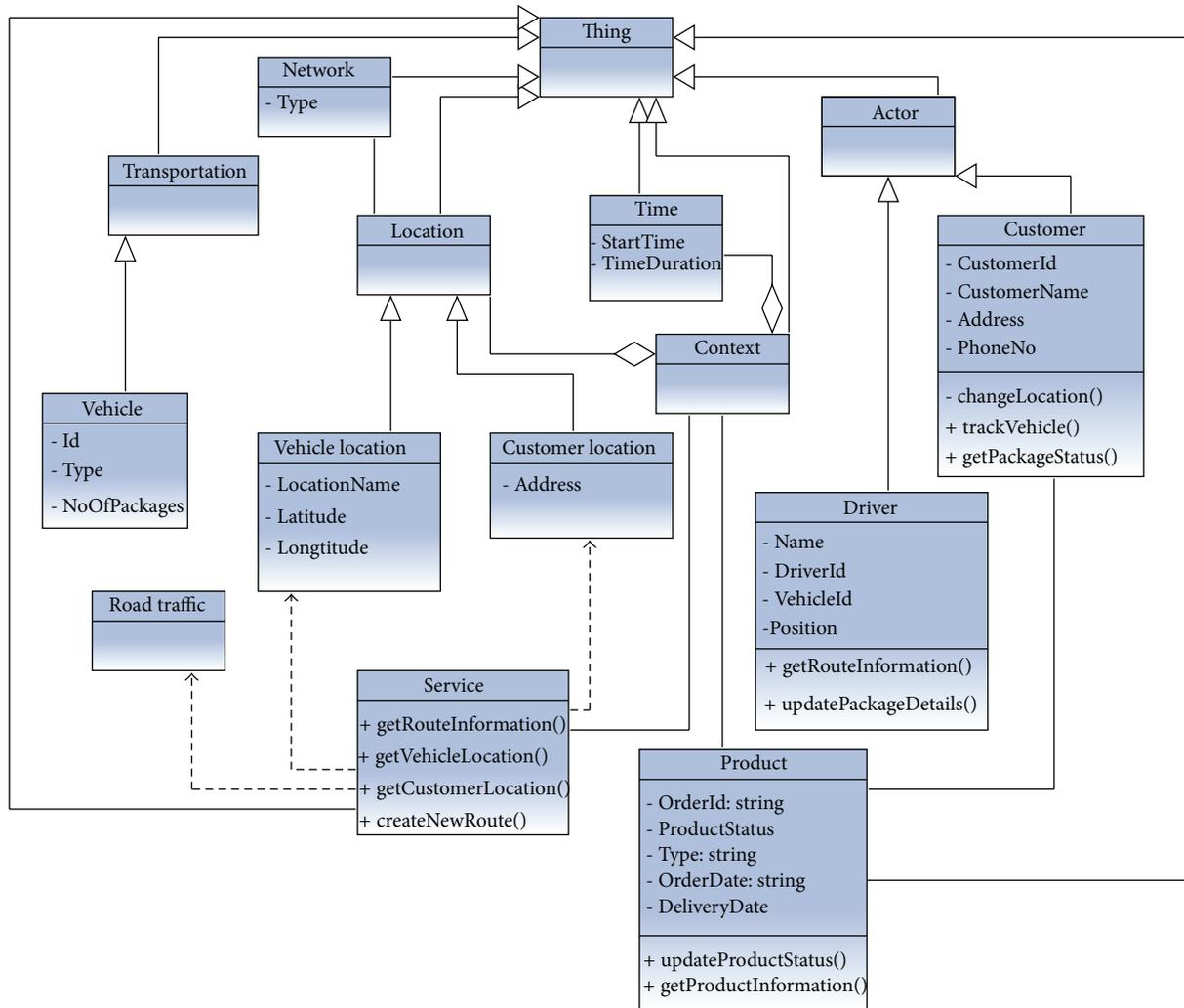


FIGURE 6: Class diagram of logistics ontology.

B's location should come later than the flow of time context and location context of Customer A or C in the new route which takes into account the time information of Customer A in 1, the new route is decided as  $C \rightarrow B \rightarrow A$  or  $C \rightarrow A \rightarrow B$ , and the newly decided route change recommendation information is indicated in real time on the smart device of the person in charge.

- (4) The service engine of the server tracks and monitors in real time the logistics/delivery service process currently in progress.
- (5) If the current location information of Customer B is close to the moving route of the person in charge, the service engine of the server sends a message to Customer B's smart device and provides automatic SMS service so that whether or not the goods were received can be checked midway. At this time, if Customer B reads the SMS text sent to his smart device and changes the reservation to receive the goods

midway, the new details are immediately delivered to the person in charge.

- (6) Customer C is currently moving to his home and Customer C's location information is sent to the server in the method explained in 2. At this time, if Customer C's location is before the current location of B or is closer to the route of the person in charge, the new route is decided as  $C \rightarrow B \rightarrow A$  or  $C \rightarrow A \rightarrow B$ , and the newly decided route change recommendation information is indicated in real time to the smart device of the person in charge.
- (7) The final available route may be  $B \rightarrow C \rightarrow A$  (if Customer C approaches a person in charge Y before they reach Customer B) or  $C \rightarrow B \rightarrow A$  (if Customer C can approach the person in charge Y after they reach Customer B).

6.2. *Logical View of the Intelligent Logistics.* In the basic scenario, the information of the delivery item, the time and

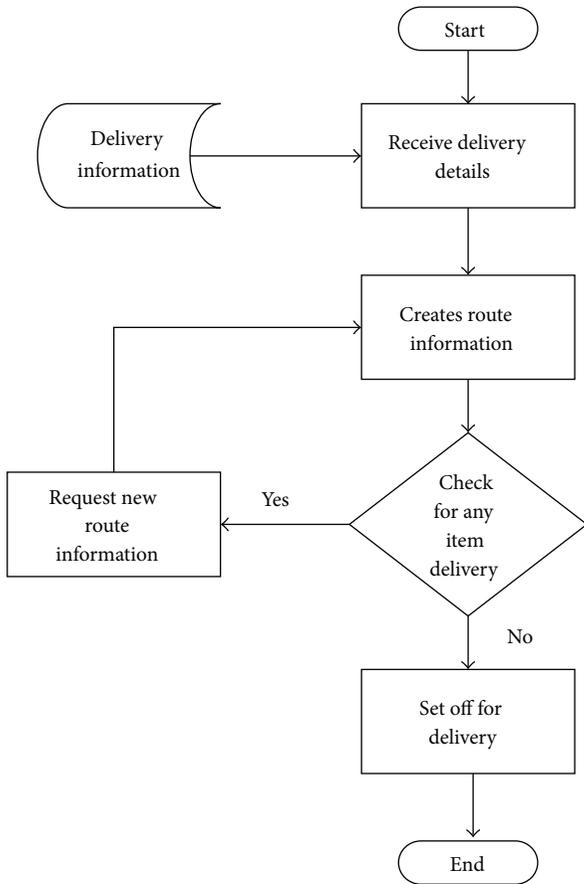


FIGURE 7: Basic scenario logic for delivery route information.

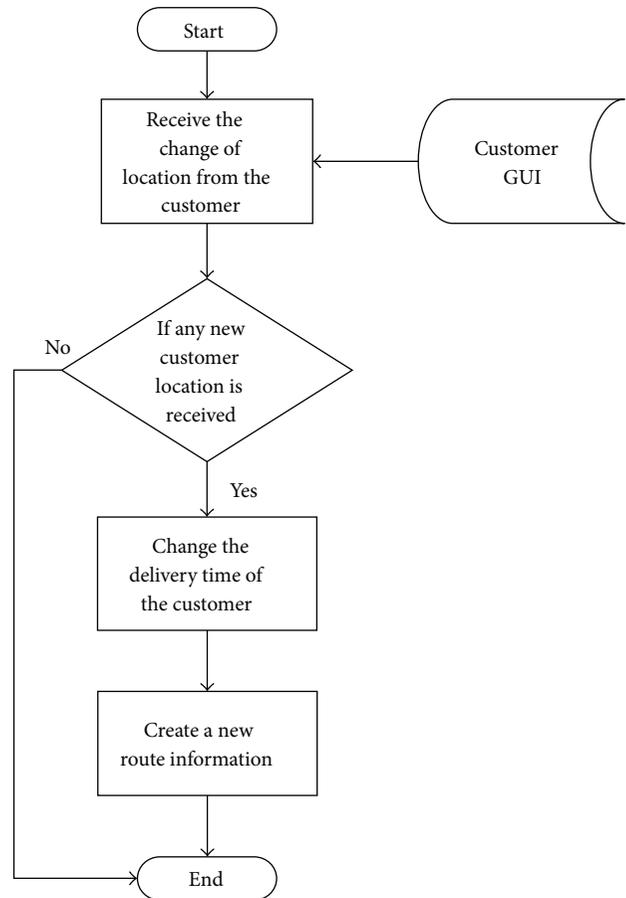


FIGURE 8: Time change request scenario logic from customer.

the location of the customer, and the optimal route for the delivery are obtained as shown in the Figure 7.

As soon as the Customer A communicates the change in time, the information is objectified as new status information. A new schedule or route is created considering the customer's needs. The updated delivery information is sent to the deliverer which is automatically displayed on the smart device of the person in charge. The approximate time change is also notified to the customers simultaneously.

The flow of the service is explained in Figure 8. The scenario mentioned in the above section was drawn with respect to the customer's change of location in which they send their own location themselves. On the other hand, the location can also be updated automatically to the server which can also be adopted by the customer.

The location information of each customer can be sent to the server through direct input, and, because the proposed smart logistics/delivery service client is already installed and running on Customers A, B, and C's smart devices, the customers' location information can be sent to the server in real time through various sensors in the smart space and the networking between GPS and customer's smart device. The details examined through the exercise scenario are a consideration of only one instance regarding changes in a very vast and complicated real time change in status

information that may arise in an actual logistics/delivery environment.

We also encountered a point in the earlier scenario, where the customer wishes to receive the delivery midway knowing the vehicle's location. When the GPS tracking is enabled on both the receiver and deliverer's smart phones, when the two devices come near to each other, the information is popped up on the customer's smartphone. If they choose to receive the package, the time change and the route information are updated and sent to the deliverer. Figure 9 shows the process flow of customer location change.

To be exact, when the distance between the customer and the vehicles is less than or equal to a kilometer, then an SMS is sent to the customer regarding the vehicle's location. A fragment of the algorithm for the above mentioned flowchart is discussed as follows:

Input Order  $O = \{O_1, \dots, O_N\}$

Matching each order with location respectively, the Order result  $OR = \{(O_1, L_{cus1}), \dots, (O_N, L_{cusN})\}$

If  $D(L_{cus8}, L_{veh}) \leq 1 \text{ km}$

Then Send SMS to customer.

...

If cus8 change the location  $L_{cus8}$ ,

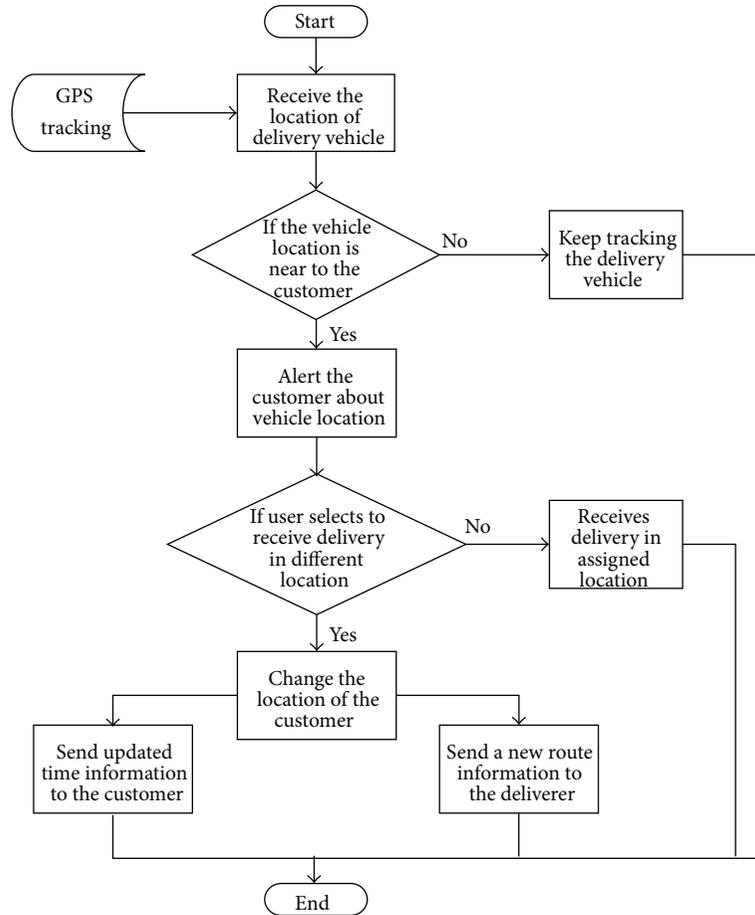


FIGURE 9: Location change request scenario from customer.

Then rearrange the sequence of order delivery.

If any of the distance from  $D(L_{cus1}, L_{veh}), \dots, D(L_{cusN}, L_{veh}) < 1$  km,

Then insert the Order ( $O_8, L_{cus8}$ ) Next to it.

Calculate the time and update the database.

$L_{cus}$  = Location of the Customer

$L_{veh}$  = Location of the vehicle

$D(L_{cus}, L_{veh})$  = Distance between the Customer Location and Vehicle Location.

Similarly the event and action based flow also helps in the situational decision making in the logistic process. With the help of intelligent logistic services through the smart phone, the customer based services are accomplished.

Context awareness plays an important role in defining the relationship, which eventually helps in the automation process. The logistic ontology was implemented in OWL with protégé 4.2. The pellet reasoner is used to verify the relationship and rules of the individual created in the logistic ontology. An instantiation of the logistic ontology is shown in the Figure 10.

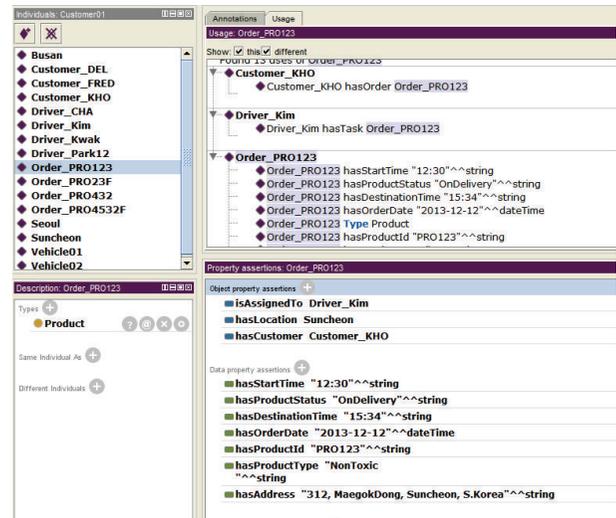


FIGURE 10: Instantiation of logistics ontology.

## 7. Conclusion and Future Work

In this paper, a user-centric smart logistics model using smart devices was proposed. The proposed service model can

generate high level status information through a rule-based context model from low level terminal sensing information arising from various sensors existing in smart space and RFID/USN and GPS and provide status information based intelligent logistics/delivery service using this. Therefore, the service model proposed in this paper is anticipated to have high availability as a diverse, mutually interactive logistics/delivery service model based on smartphones and expected to be of great help in developing related application services and systems in a ubiquitous and an intelligent computing environment which is to come in the future.

## Conflict of Interests

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

## Acknowledgments

This paper is extended and improved from accepted paper of KCIC-2013/FCC-2014 conferences. This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the CITRC (Convergence Information Technology Research Center) support program (NIPA-2014-H0401-14-1008) supervised by the NIPA (National IT Industry Promotion Agency).

## References

- [1] R. Jedermann and W. Lang, "The benefits of embedded intelligence—tasks and applications for ubiquitous computing in logistics," in *The Internet of Things*, vol. 4952 of *Lecture Notes in Computer Science*, pp. 105–122, Springer, Berlin, Germany, 2008.
- [2] P. Klaus, "Logistics research: a 50 years' march of ideas," *Logistics Research*, vol. 1, no. 1, pp. 53–65, 2009.
- [3] G. M. Giaglis, I. Minis, A. Tatarakis, and V. Zeimpekis, "Minimizing logistics risk through real-time vehicle routing and mobile technologies: Research to date and future trends," *International Journal of Physical Distribution & Logistics Management*, vol. 34, no. 9, pp. 749–764, 2004.
- [4] K. R. Prasanna and M. Hemalatha, "RFID GPS and GSM based logistics vehicle load balancing and tracking mechanism," *Procedia Engineering*, vol. 30, pp. 726–729, 2012.
- [5] E. Ferrer, A. Camacho-Martinez, L. Cardona-Hernandez, A. Machin-Cruz, L. Santos-Velez, and V. Torres-Ortiz, "The impact of mobile technology on organizational communication: rethinking the social presence theory," *Continental Journal of Information Technology*, vol. 6, no. 2, 2013.
- [6] R. Leuschner, F. Charvet, and D. S. Rogers, "A meta-analysis of logistics customer service," *Journal of Supply Chain Management*, vol. 49, no. 1, pp. 47–63, 2013.
- [7] N. Baumgartner, W. Gottesheim, S. Mitsch, W. Retschitzegger, and W. Schwinger, "BeAware!—Situation awareness, the ontology-driven way," *Data and Knowledge Engineering*, vol. 69, no. 11, pp. 1181–1193, 2010.
- [8] J. C. Augusto, V. Callaghan, A. Kameas, and I. Satoh, "Intelligent environments: a manifesto," *Human-Centric Computing and Information Sciences*, vol. 3, article 12, 2013.
- [9] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [10] B. Scholz-Reiter, W. Echelmeyer, H. Halfar, and A. Schweizer, "Automation of logistic processes by means of locating and analysing RFID-transponder data," in *Dynamics in Logistics*, pp. 323–327, Springer, Berlin, Germany, 2011.
- [11] I. Park, Y. G. Kim, S. H. Kim, C. B. Sim, and C. S. Shin, "Design and implementation of driver service system for logistics supporting vehicles," *Korean Society for Internet InFormation*, vol. 9, no. 2, pp. 197–200, 2008.
- [12] Y. M. Lee, K. W. Nam, and K. Tyu, "Design and implementation of event notification system for location-and RFID-based logistics environment," *Journal of Korea Information Processing Society D*, vol. 15, no. 5, pp. 599–608, 2008.
- [13] S. H. Lee, C. Y. Lee, D. S. Kim et al., "Sensor network deployment for warehouse management system based on RFID," *Korea Information Science Society Journal C*, vol. 14, no. 1, pp. 22–30, 2008.
- [14] D. H. Seo and I. Y. Lee, "A study on RFID system with secure service availability for ubiquitous computing," *Journal of Information Processing Systems*, vol. 1, no. 1, pp. 96–101, 2005.
- [15] G. Peng, K. Zeng, and X. Yang, "A hybrid computational intelligence approach for the VRP problem," *Journal of Convergence*, vol. 4, no. 2, 2013.
- [16] L. Yan, O. Hoerber, and Y. Z. Chen, "Enhancing Wi-Fi fingerprinting for indoor positioning using human-centric collaborative feedback," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, pp. 1–23, 2013.
- [17] A. Subramanian, L. M. A. Drummond, C. Bentes, L. S. Ochi, and R. Farias, "A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery," *Computers and Operations Research*, vol. 37, no. 11, pp. 1899–1911, 2010.
- [18] A. W. Ter Mors, J. Zutt, and C. Witteveen, "Context-aware logistic routing and scheduling," in *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS '07)*, pp. 328–335, September 2007.
- [19] V. Q. Son, B. L. Wenning, A. Timm-Giel, and C. Görg, "A model of wireless sensor networks using context-awareness in logistic applications," in *Proceedings of the 9th International Conference on Intelligent Transport Systems Telecommunications (ITST '09)*, pp. 2–7, October 2009.
- [20] S. Haseloff, *Context awareness in information logistics [Ph.D. thesis]*, TU, Berlin, Germany, 2005.
- [21] D. Werth, A. Emrich, and A. Chapko, "An ecosystem for user-generated mobile services," *Journal of Convergence*, vol. 3, no. 4, pp. 35–40, 2012.
- [22] J. V. Chen, D. C. Yen, and K. Chen, "The acceptance and diffusion of the innovative smart phone use: a case study of a delivery service company in logistics," *Information and Management*, vol. 46, no. 4, pp. 241–248, 2009.
- [23] M. Linke, "Impact of global hyperconnectivity and increased smartphone usage on the delivery and structure of IT organization in transport logistics," *International Journal of Applied Logistics*, vol. 4, no. 2, pp. 18–33, 2013.
- [24] R. Y. Shtykh and Q. Jin, "A human-centric integrated approach to web information search and sharing," *Human-Centric Computing and Information Sciences*, vol. 1, pp. 1–37, 2011.
- [25] J. K. Yin, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.

- [26] N. Watthanawisuth, N. Tongrod, T. Kerdcharoen, and A. Tuantranont, "Real-time monitoring of GPS-tracking tractor based on ZigBee multi-hop mesh network," in *Proceedings of the 7th Annual International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON '10)*, pp. 580–583, May 2010.
- [27] X. Lin and X. Zheng, "A cloud-enhanced system architecture for logistics tracking services," in *Proceedings of the International Conference on Computer, Networks and Communication Engineering (ICCNCE '13)*, 2013.
- [28] A. H. Ho, Y. H. Ho, K. A. Hua, R. Villafane, and H. C. Chao, "An efficient broadcast technique for vehicular networks," *Journal of Information Processing Systems*, vol. 7, no. 2, pp. 221–240, 2010.
- [29] M. S. Kim, J. W. Oh, Y. J. Lee, and J. S. Chae, "A design of transportation management system using smartphones," *Korea Computer Congress of Korean Institute of Information Scientists and Engineers*, vol. 37, no. 1, pp. 212–216, 2010.
- [30] A. Howard and E. Cambria, "Intention awareness: improving upon situation awareness in human-centric environments," *Human-Centric Computing and Information Sciences*, vol. 3, no. 9, 2013.
- [31] S. S. Yau, D. Huang, H. Gong, and H. Davulcu, "Situation-awareness for adaptive coordination in service-based systems," in *Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC '05)*, pp. 107–112, July 2005.
- [32] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

## Research Article

# Numeric Analysis for Relationship-Aware Scalable Streaming Scheme

Heung Ki Lee,<sup>1</sup> Jaehee Jung,<sup>1</sup> Kyung Jin Ahn,<sup>1</sup> Hwa-Young Jeong,<sup>2</sup> and Gangman Yi<sup>3</sup>

<sup>1</sup> Samsung Electronic Co., Suwon, Republic of Korea

<sup>2</sup> Humanitas College, Kyung Hee University, Seoul, Republic of Korea

<sup>3</sup> Department of Computer Science & Engineering, Gangneung-Wonju National University, Gangwon-do, Republic of Korea

Correspondence should be addressed to Gangman Yi; [gangman@cs.gwnu.ac.kr](mailto:gangman@cs.gwnu.ac.kr)

Received 13 March 2014; Accepted 29 April 2014; Published 12 June 2014

Academic Editor: Young-Sik Jeong

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

Frequent packet loss of media data is a critical problem that degrades the quality of streaming services over mobile networks. Packet loss invalidates frames containing lost packets and other related frames at the same time. Indirect loss caused by losing packets decreases the quality of streaming. A scalable streaming service can decrease the amount of dropped multimedia resulting from a single packet loss. Content providers typically divide one large media stream into several layers through a scalable streaming service and then provide each scalable layer to the user depending on the mobile network. Also, a scalable streaming service makes it possible to decode partial multimedia data depending on the relationship between frames and layers. Therefore, a scalable streaming service provides a way to decrease the wasted multimedia data when one packet is lost. However, the hierarchical structure between frames and layers of scalable streams determines the service quality of the scalable streaming service. Even if whole packets of layers are transmitted successfully, they cannot be decoded as a result of the absence of reference frames and layers. Therefore, the complicated relationship between frames and layers in a scalable stream increases the volume of abandoned layers. For providing a high-quality scalable streaming service, we choose a proper relationship between scalable layers as well as the amount of transmitted multimedia data depending on the network situation. We prove that a simple scalable scheme outperforms a complicated scheme in an error-prone network. We suggest an adaptive set-top box (AdaptiveSTB) to lower the dependency between scalable layers in a scalable stream. Also, we provide a numerical model to obtain the indirect loss of multimedia data and apply it to various multimedia streams. Our AdaptiveSTB enhances the quality of a scalable streaming service by removing indirect loss.

## 1. Introduction

The motivation for this paper is to provide high-quality multimedia service over mobile networks. In a mobile network, two trends make it difficult to improve multimedia service. First, the introduction of smart phones has dramatically increased the volume of video traffic over mobile networks [1], with video consuming most of the available wireless resources [2]. Second, users expect a high-quality streaming service. Thus additional wireless resources are required to satisfy those users [1].

In this paper, we present a solution for enhancing the quality of streaming services over mobile networks. One solution is to improve the capacities of wired and wireless links between the multimedia streaming server and

the mobile client. However, updating the mobile network infrastructure is too expensive. Even though Internet Service Providers (ISPs) have continued to improve the speed of mobile networks, they cannot satisfy user thirst for high-quality multimedia services.

Another solution is to decrease the error rate of mobile networks. Streaming services over mobile networks deliver media data under error-prone network environments [3, 4]. Also, users of mobile networks compete for limited wireless resources for receiving multimedia streams. Such severe competition dramatically increases the error rate of mobile networks. Therefore, the quality of the streaming service might be reduced by the increased error rate of mobile networks. However, we cannot control cross-traffic from other devices. In this paper we explore a third approach:

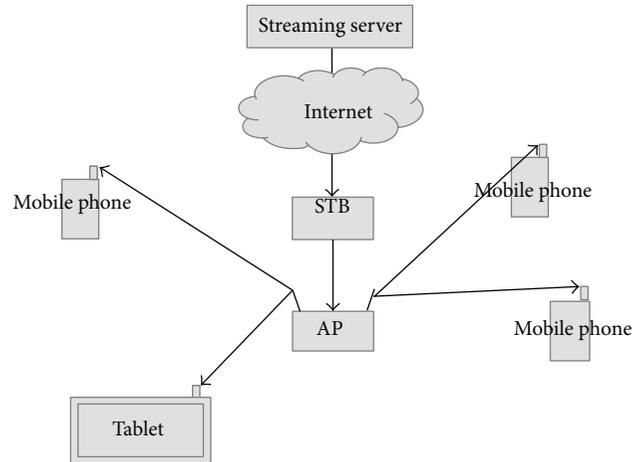


FIGURE 1: Diagram of scalable streaming in a wireless environment.

decreasing the negative effect resulting from losing packets. Error propagation through direct or indirect lost packets worsens the quality of streaming services over mobile networks. As a result, the high quality of media data in mobile networks can lead to frustrating user experience because of frequent data buffering or distorted frames.

To address this problem, content providers (CPs) calibrate streaming and cache servers through a scalable streaming scheme depending on network status. If the CP provides media at different qualities without a scalable streaming scheme, they would need to store all these different media on their own servers and incur costs associated with maintenance of redundant media of different quality. This would increase the cost of maintaining media streams [5–8]. However, a scalable streaming service divides one large media file into several layers. Therefore, by using a scalable streaming scheme, CPs can eliminate redundant stored media data. A scalable streaming scheme consists of a base layer and enhancement layers. The base layer is necessary for decoding; enhancement layers are not themselves decoded but they increase the quality of the streaming service. In a congested network, the mobile node just requests the base layer for seamless streaming. When the mobile network is stabilized, the CP provides all scalable layers including the base layer and enhancement layers to the user. Therefore the user can obtain streaming service with high quality.

Figure 1 shows an example of a scalable streaming service for a mobile network. One set-top box (STB) receives the scalable streams from the streaming server and then forwards scalable layers to mobile nodes including mobile phones and tablets. Usually, the STB is located at one place (e.g., a restaurant, a shop, or a bus station), so the network between the streaming server and the STB is a stable wired network that provides the connection without delay or lost packets during the streaming service. However, the wireless network between the STB and the mobile node is not guaranteed. In wireless networks, several mobile nodes share wireless resources for providing service to mobile users. Interference between mobile nodes can cause the network to drop or delay packets.

Regardless of the benefit of a scalable streaming service, the relationship between layers degrades the quality of the scalable streaming service over an error-prone network. Even though one layer is transmitted successfully, the absence of a reference layer wastes other related layers at the decoder. Therefore, the loss of one packet invalidates its own layer and its referring layer. To improve the performance of scalable streaming services over error-prone networks, we should reduce the dependency between layers, thereby decreasing the amount of related media data for one packet and wasted media data caused by single-packet loss. We suggest an *adaptive set-top box (AdaptiveSTB)* that lessens the dependency between layers transmitted over wireless networks. The *AdaptiveSTB* is located between the wired network and the wireless network and converts complex hierarchical scalable streams into scalable streams consisting of layers with low dependency.

In summary, in this paper we provide the following contributions. We present a service design for an *AdaptiveSTB* that decreases the dependence among scalable layers. Our *AdaptiveSTB* converts the receiving scalable streams with high dependency into scalable streams with low dependency. As a result, it decreases the indirect loss of media data and increases streaming service performance even over mobile networks. We then analyze a media scheme to convert scalable streams. Also we provide a numerical model for showing the amount of multimedia data. Finally, we apply our *AdaptiveSTB* to various streams. In Section 2, we introduce existing adaptive scheme and scalable scheme in detail. Section 3 explains our *AdaptiveSTB* in detail. Section 4 shows experiments for scalable streaming service and results, while Section 5 concludes this paper.

## 2. Related Work

### 2.1. Background

**2.1.1. Scalable Streaming versus Adaptive Streaming.** There are two schemes for adapting the quality of multimedia stream

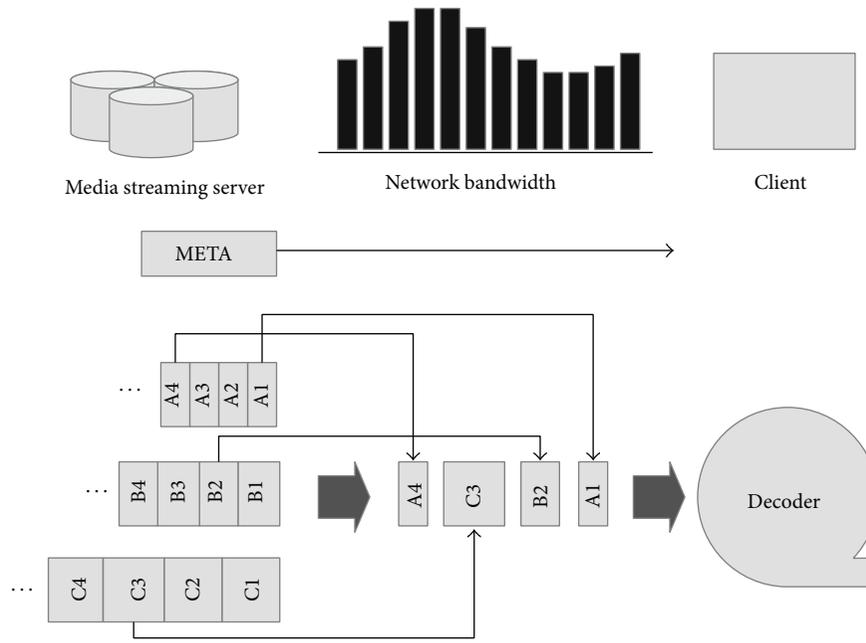


FIGURE 2: Adaptive streaming scheme.

services based on network status: *adaptive streaming* and *scalable streaming*.

In an adaptive streaming scheme, redundant multimedia streams with different quality data are stored in a storage area. Based on the bandwidth, adaptive streaming schemes can switch which stream to send to the user. Figure 2 shows a general adaptive streaming scheme. When the user initiates the adaptive streaming service, the user requests the metafile that describes which streams are available to the user. When the network is congested at the beginning of the streaming service, the adaptive streaming scheme selects the A1 media stream, which has the lowest quality. When the network condition becomes stable, the adaptive streaming scheme switches to the B2 media stream, which is of medium quality. When the network allows for higher quality streams, the user can request the C3 media stream, which has the highest quality. After that, the user changes to media streams with the lowest quality when the network is congested.

Figure 3 shows the scalable streaming scheme. With a scalable streaming service, one stream can be divided into several layers. The base layer can provide the streaming service by itself, but the quality is improved when more layers are included. Upon first use, the user requests the metafile that shows which multimedia streams can be served. The user selects the lowest stream that includes only the A1 base layer. When the network becomes stable, the user selects more layers with medium quality to add B2 enhancement layers to A2 base layers. When the available bandwidth of the network is approved for the highest quality of scalable streaming service, the user requests whole layers including one base layer A3 and two enhancement layers B3 and C3. The user only requests base layer A4 to save wireless resources over a congested network. The scalable streaming service provides the proper quality of the streaming service based on

network conditions. Also, service providers can save space for storing media layers, thereby reducing the cost of maintaining the multimedia system.

**2.1.2. Relationship between Frames.** One media file has various frames, each of which shows one scene in the stream. There are three kinds of frames in the stream: the I frame, the P frame, and the B frame. The I frame contains all the information for showing one scene, whereas a decoder needs to be used to get additional information from other frames for decoding P or B frames. The P frame requires some information from the previous P or I frame, whereas the B frame needs to obtain information from the previous P or I frame and the future P or I frame at the same time.

The hierarchical structure between frames is critical to determining the quality of the scalable streaming service. The relationship between layers determines which layer can be available at the decoder. The scalable stream extracts multiple layers from one stream following each policy. The referring layer cannot be decoded without the reference layer. Therefore, the scalable stream increases the dependency between layers, adding an interframe relationship, thereby complicating the relationship between layers and making them harder to decode.

Figure 4 shows the relationship between layers in *MP4 scalable streaming*. There are several scalable layers: the base layer (Layer 1) and several enhancement layers (Layer 2, Layer 3, and Layer 4). Layer 1 is required for decoding the frame; Layer 2 improves the quality of Layer 1. Therefore, when Layer 1 is not available, Layer 2 cannot be decoded. Also, Layer 3 and Layer 4 are required above the scalable layers for each frame. The I frame can be decoded by itself, but the P frame refers to one previous frame. In Figure 4, only when Layer 1 of

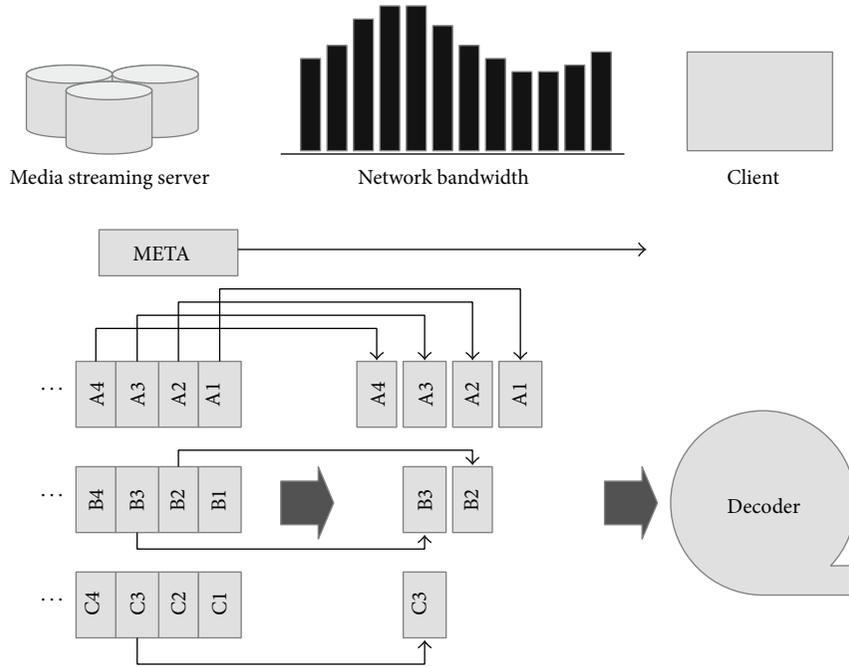


FIGURE 3: Scalable streaming scheme.

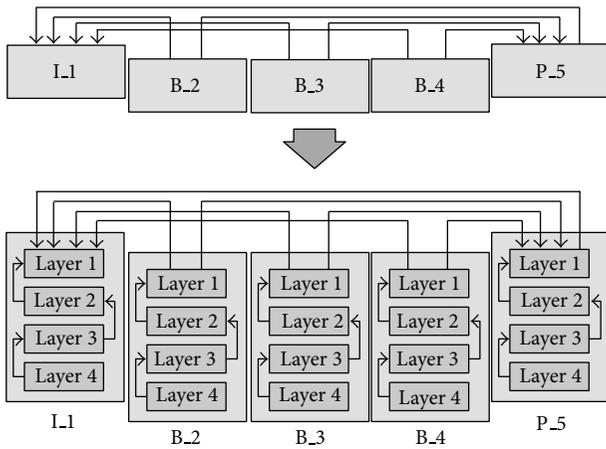


FIGURE 4: MP4 scalable streaming scheme.

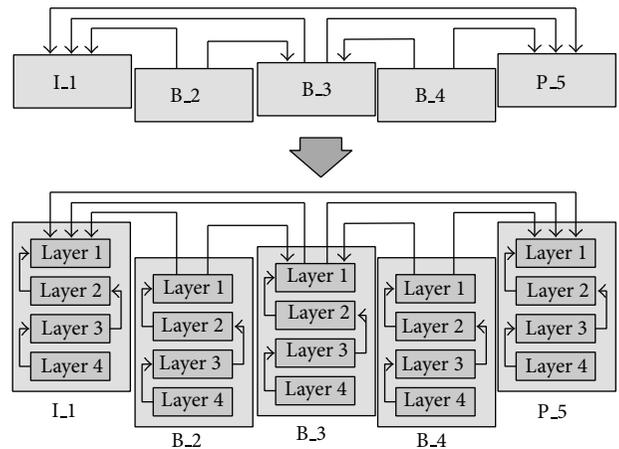


FIGURE 5: H.264 scalable streaming scheme.

the I<sub>1</sub> frame is available can scalable layers of the P<sub>5</sub> frame be decoded. Also, the B frame requires two frames; it needs Layer 1 of the I<sub>1</sub> frame and Layer 1 of the P<sub>5</sub> frame.

In Figure 5, the I and P frames in *H.264 scalable streaming* require no frames or one frame for decoding a frame like MP4 scalable streaming. Also, the relationship between scalable layers at two frames is similar to that in MP4 scalable streaming, but B frames in H.264 scalable streaming require two or more frames to be decoded. In Figure 5, the B<sub>2</sub> frame requires three scalable layers: Layer 1 of the I<sub>1</sub> frame, Layer 1 of the P<sub>5</sub> frame, and Layer 1 of the B<sub>3</sub> frame. This complicated hierarchical structure between frames decreases the network's availability for providing a temporal scalable streaming service.

In this paper, we propose an *AdaptiveSTB* that improves the performance of scalable streaming by reducing the complexity of the relationships between layers. The *AdaptiveSTB* receives the media data from the media server and forwards them to the client through a wireless network. The *AdaptiveSTB* converts the H.264 scalable stream into an MP4 scalable stream before forwarding the cached media.

## 2.2. Previous Work

**2.2.1. Streaming Service.** Numerous schemes have been proposed for handling partial errors in packets. To enhance the quality of a scalable streaming service, [9] increases the availability of the base layer through Multiple Description

Coding (MDC). [1] suggests using the MDC scheme for devices with low computing power and narrow dynamic available bandwidth. Unequal Error Protection (UEP) in layer partitioning has been suggested to improve the performance of streaming in [10]. [11] adjusts the level of Forward Error Correction (FEC) for providing a scalable streaming service. [12] suggests using SoftCast to provide unequal error protection in the video encoding step in wireless networks. In [13], FlexCast selects critical bits of a video through distortion grouping for providing efficient video encoding schemes. [14] suggests a scheme in which a peer device forwards the receiving layers to other devices.

Also, there has been much research on transcoding schemes. ISP proxies, a task dispatcher, and a client provide the transcoding scheme through multiple caching policies in [15]. [16] suggests using Hadoop to conduct a transcoding scheme for a variety of video content suitable under network conditions. In [17], CloudStream is used to enhance the performance through a parallel scheme in transcoding videos. [18] has evaluated the resource demand for a transcoding scheme in various media services.

*2.2.2. Use of Intermediate Nodes for Streaming Video.* For improving streaming service quality, active intermediate nodes have been deployed during streaming [19–22]. When a network is congested, intermediate node degrades quality of the cached stream and then provides it to the mobile node in [19, 23].

In [20, 24], an intermediate node removes the streaming data with large jitter. The intermediate node decides to retransmit the lost packet using the presentation time of the multimedia data in [21]. [22] suggests a scheme in which the intermediate node sends the lost section of multimedia to the user when the user is missing some section in the broadcasting service.

A set-top box is an intermediate node located between the wired network and the wireless network through the streaming service. In [25], the STB consists of four blocks: a Media Codec, a Graphic Module, a Presentation Module, and a Network Module. [26] provided additional functions to the TV STB including video recording and adapting quality of recorded video. [27] detected the lost packets and jitter for improving service to the user. [28, 29] propose using video proxies to increase quality of the streaming service. Also, [30, 31] improve the performance through caching and prefetching strategies.

### 3. Adaptive Set-Top Box

*3.1. Simple Scalable Streaming Service.* The quality of a scalable streaming service is influenced by the dependency among the scalable layers in a scalable streaming service. The hierarchical relationship between scalable layers determines the decoding possibilities for transmitting the packet to the client over the wireless network. When reference frames are not transmitted successfully, the referring frames cannot be decoded. The complicated reference relationships between scalable layers of H.264 streaming increases the possibility

TABLE 1: Scalable streaming conversion variables.

|               |                                                                                                                                     |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------|
| $G_g$         | The $g$ th GoV in the stream                                                                                                        |
| $F_{g,f}$     | The $f$ th frame at the $g$ th group of video (GoV)                                                                                 |
| $RF(j, g, f)$ | The $j$ th reference frame of the $f$ th frame at the $g$ th GoV                                                                    |
| $L_{g,f,k}$   | The $k$ th layer of the $f$ th frame at the $g$ th GoV                                                                              |
| $P_{g,f,k,s}$ | The $s$ th packet in the $k$ th layer of the $f$ th frame at the $g$ th GoV                                                         |
| NG            | Number of GoVs in the multimedia stream                                                                                             |
| $NF_g$        | Number of frames at the $g$ th GoV                                                                                                  |
| $NRF_g$       | Number of reference frames of the $f$ th frame at the $g$ th GoV                                                                    |
| $NL_{g,f}$    | Number of layers of the $f$ th frame at the $g$ th GoV                                                                              |
| $NP_{g,f,k}$  | Number of packets in the $k$ th layer of the $f$ th frame at the $g$ th GoV                                                         |
| PLR           | Packet loss rate                                                                                                                    |
| $VLI_{g,f,k}$ | Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV                                                                  |
| $VLV_{g,f,k}$ | Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV under the relationship between scalable layers in the same frame |
| $VLH_{g,f,k}$ | Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV in the stream                                                    |

of discarding the referring frame. Figures 4 and 5 show the relationship between scalable layers in a scalable streaming scheme. In MP4 scalable streaming, whole B frames require two frames, including an I frame or a P frame. Even though other B frames are dropped, the transmitted B frame can be decoded. When B.3 and B.4 frames are dropped, the B.2 frame can be decoded. However, in H.264 scalable streaming, a complicated relationship exists between B frames. When B.3 and B.4 frames are dropped, the B.2 frame cannot be decoded.

*3.2. AdaptiveSTB.* Our *AdaptiveSTB* decreases the dependency among scalable layers in scalable streaming, thereby enhancing the performance of the streaming service in wireless environments. For the decoding layer in a scalable stream, the decoder needs to obtain information from other reference layers and know the dependency between scalable layers. When reference layers are lost, the decoder discards the referring layers. For enhancing the quality of a scalable streaming service, it is critical to decrease such indirect loss. Our *AdaptiveSTB* converts H.264 scalable streaming into MP4 scalable streaming before transmitting layers over the wireless network.

When whole packets are available, the layer can be decoded in the scalable stream. When one packet is lost, other data in the layer cannot be used for the decoding. Therefore, validation of the layer can be assured only when all its packets are available. Based on the terms in Table 1, the validity of the scalable layer is given by

$$VLI_{g,f,k} = \prod_{s=1}^{NP_{g,f,k}} (1 - PLR(P_{g,f,k,s})). \quad (1)$$

TABLE 2: Scalable streams.

| Layer | QP   | Frame rate | Frame size |
|-------|------|------------|------------|
| 0     | 38.0 | 15         | 320 × 240  |
| 1     | 32.0 | 30         | 320 × 240  |
| 2     | 30.0 | 30         | 320 × 240  |
| 3     | 28.0 | 30         | 640 × 480  |
| 4     | 26.0 | 30         | 640 × 480  |

When the error rate of the wireless link increases, most discarded scalable layers do not satisfy this equation. One frame is divided into several layers, so the reference layer is required to decode the referring scalable layer in the scalable streaming service. The number of scalable layers available is based on the vertical dependency among scalable layers. The validity of the scalable layer is given by

$$VLV_{g,f,k} = \prod_{j=1}^k \prod_{s=1}^{NP_{g,f,j}} (1 - \text{PLR}(P_{g,f,j,s})). \quad (2)$$

Finally, the decoder should check whether reference frames are available. The decoder does not require all the scalable layers of the reference frame to decode the referring frame. If the first layer of the reference frame is available, the reference frame can be decoded, and the validity of the scalable layer in the stream is given by

$$VLH_{g,f,k} = \prod_{j=1}^{NRF_g} (VLH_{g,RF(j,g,f),1}) \times \prod_{j=1}^k \prod_{s=1}^{NP_{g,f,j}} (1 - \text{PLR}(P_{g,f,j,s})). \quad (3)$$

When the error rate of the wireless link decreases, most of the discarded scalable layers will not satisfy this equation.

## 4. Experimental Result

For verifying the performance of our *AdaptiveSTB*, we conducted a network simulation on an NS-2 simulator [32] based on data extracted from real scalable streaming data. We downloaded five movie trailer clips and one video clip from the Internet, then generated scalable layers from them using a scalable encoder.

**4.1. Scalable Multimedia.** We used the Joint Scalable Video Model (JSVM) [33] for generating scalable layers from six H.264 streams. We created five scalable layers from several original streams. The following configuration is used for generating scalable streams. QP in Table 2 stands for a quantization parameter. This value divides pixel information at each frame. Therefore, detailed pixel information for each frame is saved when QP is small. The frame rates indicate how many frames are displayed in a second. High-frame-rate streams achieve smooth transitions between frames. The frame size gives the width and height of a scalable stream.

TABLE 3: Scalable streams.

| Frame name           | Genre      | Number of frames |
|----------------------|------------|------------------|
| Amazing Caves        | Adventure  | 2031             |
| The Bourne Ultimatum | Action     | 2125             |
| I Am legend          | Drama      | 2397             |
| Fantastic 4          | Action     | 3017             |
| Foreman              | Video Clip | 399              |
| To the Limit         | Adventure  | 919              |

A scalable stream with a large frame size can hold more pixel information.

Layer 0 is encoded at 15 frames per seconds (fps) with a QP of 38. In addition, the resolution of the base layer is suitable for a 320 × 240 display. When Layer 1 is added, the frame rates are increased to 30 fps and QP is decreased to 32. This provides a clear scene for the user. As more scalable layers become available at the scalable decoder, the quality of the scalable streaming service increases. Of the movie and video clips we used (see Table 3) for simulation, *Amazing Caves* and *To the Limit* are adventure movie, so scenes can change quickly. *The Bourne Ultimatum* and *Fantastic 4* are action movies where variance between frames is large. Scenes do not change quickly in *I Am Legend* and *Foreman*.

**4.2. Simulation Environments.** Figure 6 shows an overall diagram of our network simulation with scalable streams. For verifying the performance of our *AdaptiveSTB*, we generated real scalable layers using the JSVM codec from real media streams and then obtained type, time, and size of each frame for a scalable layer. Based on gathering frame information from scalable layers, we ran a network simulation using an NS-2 simulator. In the network simulation, a stream server transfers multimedia data based on the obtained size information from real scalable layers. The capacity of the wired connection between the stream server and the STB was 100 Mbps; the wireless nodes were connected through a 10 Mbps wireless link. We ran simulations with various error rates.

In Figure 6, the client for streaming service checks the arrival time of each incoming packet from the stream server. If the packet has already been transmitted at the obtained frame time of the multimedia data in the packet, the client for streaming service can decode the multimedia data in the packet. For example, multimedia data that should be displayed four minutes after starting play is delivered three minutes after the first multimedia data arrived. The delivered multimedia data can be decoded at the client for streaming service. However, if the multimedia data are delivered five minutes later, they are discarded.

MPEG standards recommend that the decoder skip corrupted multimedia data in the next synchronization position (e.g., *start code* or *resync code*) to reduce errors. The STB can check the received scalable layer for detectable corrupted scalable layers and then skip the corrupted multimedia data caused by other dropped or delayed packets. However, because we cannot use real multimedia data in our network

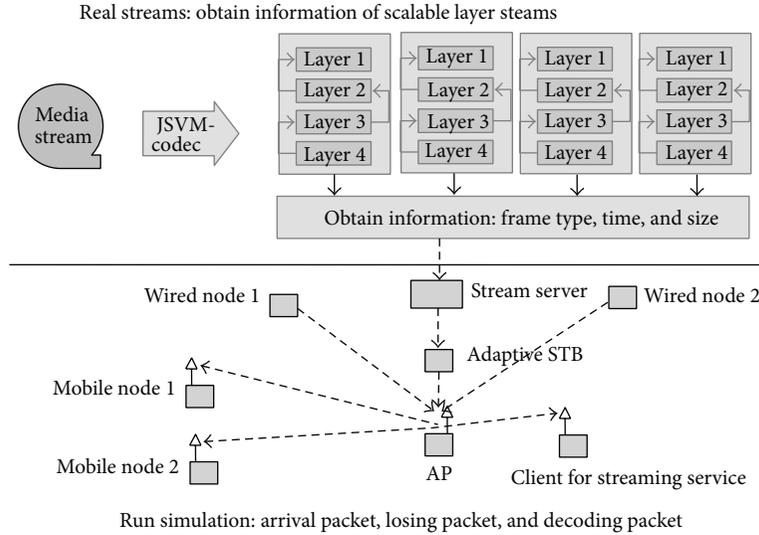


FIGURE 6: Simulation environment.

simulation, it is difficult to ascertain how much multimedia data are corrupted by lost or delayed packets.

For detecting corrupted multimedia data in the simulation, the stream server adds more information to the generating packets, including frame\_no, frame\_seq, layer\_id, and frame\_flag. Here, frame\_no stands for the order of the transmitted frame, and frame\_seq is the sequence number of the packets. Our *AdaptiveSTB* can detect lost packets using frame\_seq. The label layer\_id identifies the scalable layer as Layer 0, Layer 1, Layer 2, Layer 3, or Layer 4. Lastly, frame\_flag indicates whether the packet is the first packet (frame\_flag = 0), an intermediate packet (frame\_flag = 1), or the last packet (frame\_flag = 2) in the frame.

Figure 7 shows that each packet contains four pieces of information in the following order: frame\_no, frame\_seq, frame\_flag, layer\_id, and data. In the information, the first information shows frame\_no. Our *AdaptiveSTB* identifies packets using frame\_no, frame\_seq, and layer\_id. The decoder at the client for streaming service checks whether scalable layers are available based on additional information including frame\_no, frame\_seq, layer\_id, and frame\_flag.

### 4.3. Simulation Results

4.3.1. *Indirect Loss.* Figure 8 shows the ratio between indirect lost multimedia data and received multimedia data from the STB. In the figure, the *x*-axis is the error rate over the wireless network, and the *y*-axis is the ratio between indirect lost multimedia data and received multimedia data in the client for streaming service. The interframe encoding scheme in the MPEG standard means that some portions of the frame are referred from other frames, but this scheme increases the dependency among frames and the possibility of discarding received frames by the client for streaming service. Such a discarding of frames reduces the chance to transmit other scalable layers. In our simulation based on real scalable streams, MP4 scalable streaming outperformed

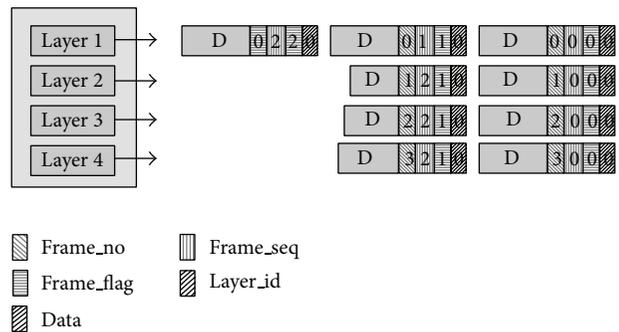


FIGURE 7: Simulation packet management.

H.264 scalable streaming. The high complexity of H.264 scalable streaming increases the number of discarded scalable layers indirectly.

*Amazing Caves* is a high-quality stream; therefore, the size of one frame is huge. It is difficult for all the packets in the frame to be delivered at the client for streaming service before decoding the frame. There is a gap between MP4 scalable streaming and H.264 scalable streaming when the low error rate over the wireless network is low, but when error rate increases, there is no difference between the two schemes. Most scalable layers do not satisfy (1), so incomplete scalable layers are discarded directly. The simulation results of *The Bourne Ultimatum* and *I Am Legend* appear to be similar to those of *Amazing Caves*. At low error rate, the ratio between indirect lost scalable layers and received scalable layers of MP4 scalable streaming is smaller than that of H.264 scalable streaming.

In *Fantastic 4*, *Foreman*, and *To the Limit*, the size of frames is relatively small. The small number of packets generated in one frame increases the possibility of decoding the scalable layer. The client for streaming service decodes scalable layers according to (3).

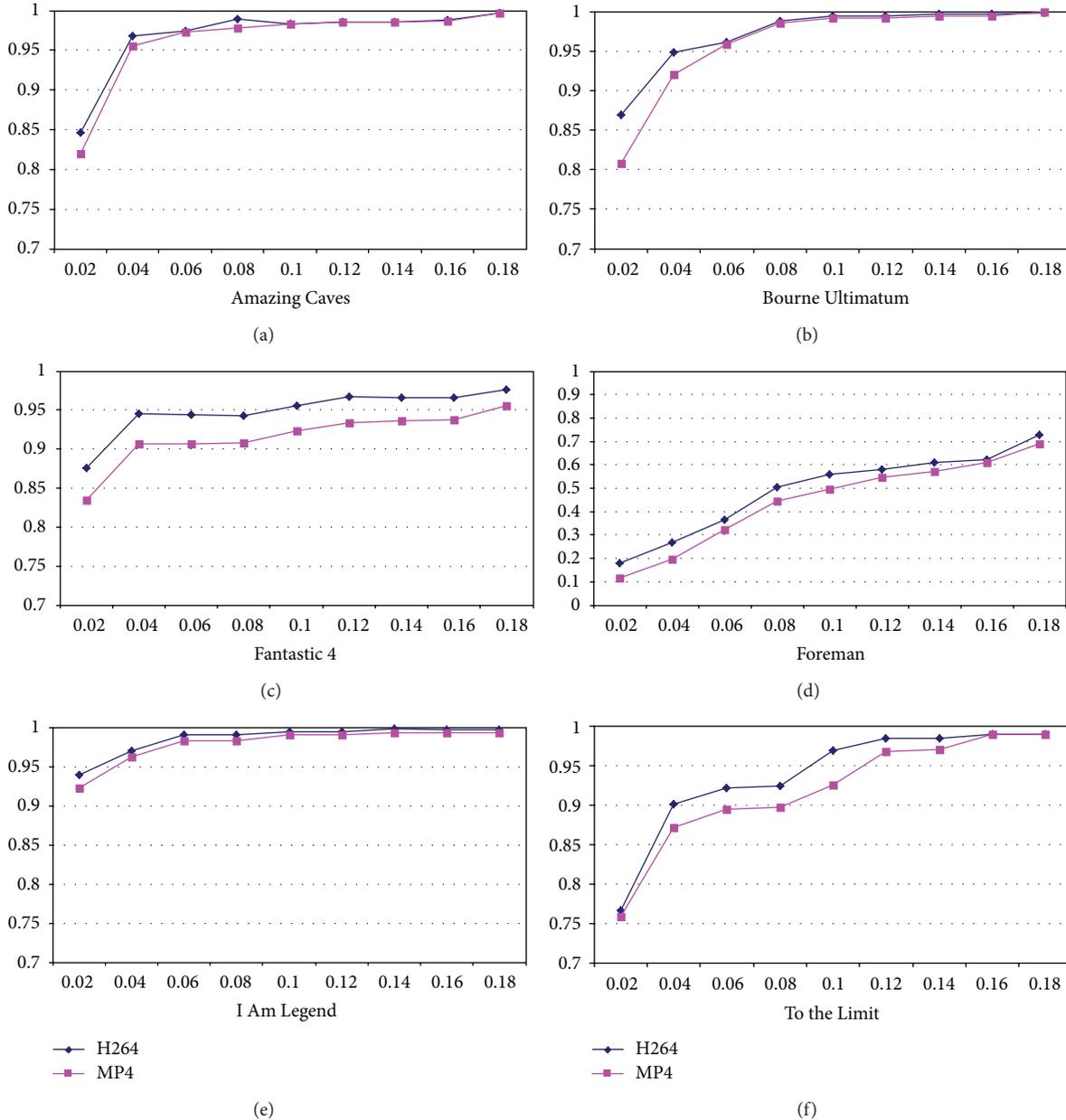


FIGURE 8: Ratio between indirect lost layers and received layers.

**4.3.2. Decoding Frame.** Figures 9 and 10 show the ratio between transmitted multimedia data and decoded multimedia data. Figure 9 shows how many bytes are displayed, and Figure 10 shows how many frames are available to the user. These two graphs show similar results. However, the ratio between B frames and I frames in the stream gives a different result.

In Figure 9, the  $x$ -axis is the error rate of the wireless network and the  $y$ -axis is the ratio between the size of the decoded multimedia data at the client for streaming service and the size of the transmitted multimedia data from the STB. High ratios mean that most of the transmitted multimedia data from the STB are decoded, so streaming services with high ratios can provide clear streams to users.

In *Amazing Caves*, MP4 scalable streaming exhibits a higher ratio than H264 scalable streaming at lower error rates, but the two schemes are similar at high error rate. Most multimedia data are dropped because they do not satisfy (1). *The Bourne Ultimatum* and *I Am Legend* exhibit similar results.

In *Fantastic 4*, *Foreman*, and *To the Limit*, MP4 scalable streaming exhibits higher ratios than H264 scalable streaming at all error rates, which means that more multimedia data are transmitted in MP4 scalable streaming. The size of frames in *Fantastic 4*, *Foreman*, and *To the Limit* are relatively smaller than those of other streams. One frame is divided into a small number of packets, so more multimedia data can satisfy (1). The multimedia data are decoded by using (3).

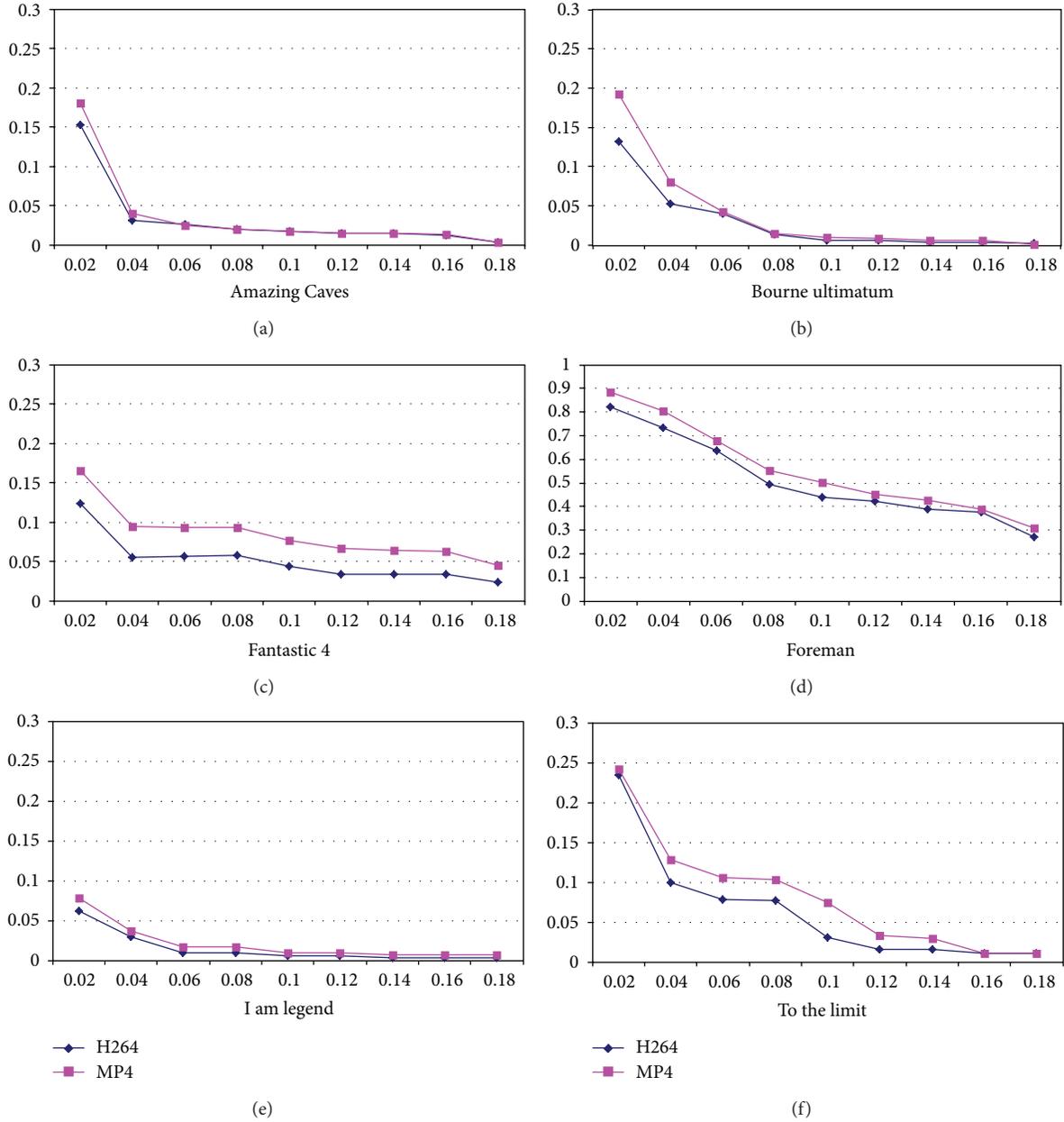


FIGURE 9: Ratio between decoded layers and sent layers (based on bytes).

Figure 10 shows the ratio between transmitted scalable layers from the STB and decoded scalable layers at the client for streaming service. The  $x$ -axis is the error rate of the wireless network and the  $y$ -axis is the ratio between decoded scalable layers at the client for streaming service and transmitted scalable layers from the STB. High ratios between decoded layers and transmitted layer indicate that many scalable layers are decoded among the transmitted scalable layers, meaning that the STB provides a good quality scalable streaming service.

In the *Amazing Caves*, a gap between MP4 scalable streaming and H264 scalable streaming is distinguishable at low error rate. However, the gap closes at high error rate

as the result of the discarding of most scalable layers. Such simulation results follow according to (1). In *The Bourne Ultimatum* and *I Am Legend*, more frames in MP4 scalable streaming are decoded than in H264 scalable streaming at low error rate, but, as error rate increases, the client for streaming service drops more receiving scalable layers following (1). In *The Bourne Ultimatum*, H264 scalable streaming is better than MP4 scalable streaming even at some high error rates. At low error rate for *The Bourne Ultimatum*, the ratio of scalable layers in Figure 10 is higher than the ratio of multimedia data in Figure 9. *The Bourne Ultimatum* contains a high ratio of B frames, so the ratio of scalable layers is increased for a small ratio of multimedia data.

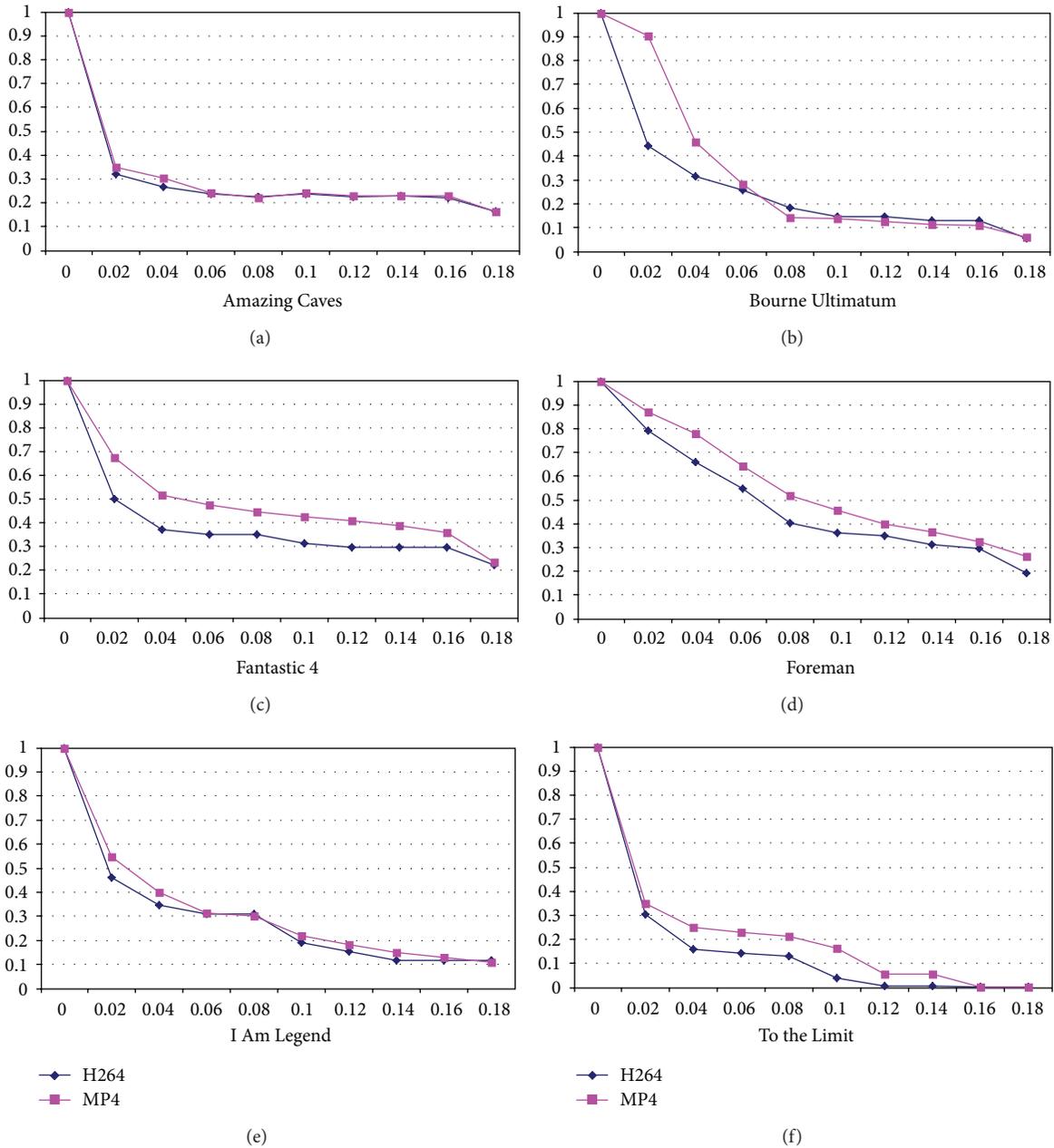


FIGURE 10: Ratio between decoded layers and sent layers (based on frames).

In *Fantastic 4*, *Foreman*, and *To the Limit*, MP4 scalable streaming outperforms H264 scalable streaming for all error rates. In those streams, one frame is divided into a small number of packets, so the scalable layers themselves are available from (1). Most of discarded scalable layers do not satisfy (3).

Our simulation results show that a scalable scheme with low dependence among scalable layers provides good service to users. At low wireless network error rates, the relationship among scalable layers determines the quality of scalable streams; this is especially critical for streams with small sized frames.

## 5. Conclusion

In this paper, we show that our *AdaptiveSTB* converts scalable layers with complicated dependency into simple scalable layers, thereby enhancing the scalable streaming service over a wireless network. We found that the main reason for a reduction in quality of scalable streaming over a wireless network had to do with the error rate. When a scalable layer can only be delivered with a high error rate, the dependency among scalable layers exerts little influence on the quality of the scalable streaming service. However, when the error rate of the wireless network is low or the size of scalable

layers is small, the quality of the scalable streaming service is determined by the dependency among scalable layers.

We perform packet-level analysis for scalable streaming service over a wireless network. Additionally, we suggest formulas for the expected quality of the scalable streaming service and prove the performance of our *AdaptiveSTB* through simulations in wireless networks. We compare the performance of scalable streams over wireless networks with various error rates. Future work will address limitation of resources (e.g., memory and computing power) at the set-top box as well as various network environments.

## Conflict of Interests

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

## Acknowledgments

This paper is a revised and extended version of a paper that was originally presented at the 2014 FTRA International Symposium on Frontier and Innovation in Future Computing and Communications. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education (2013RIA1A2063006).

## References

- [1] "Cisco-Visual-Networking-Index," Cisco Systems, 2010–2015, <http://bit.ly/pIDtBX>.
- [2] J. Erman, A. Gerber, K. K. Ramadrishnan, S. Sen, and O. Spatscheck, "Over the top video: the gorilla in cellular networks," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC '11)*, pp. 127–136, New York, NY, USA, November 2011.
- [3] J. C. Tsai and N. Y. Yen, "Cloud-empowered multimedia service: an automatic video storytelling tool," *Journal of Convergence*, vol. 4, no. 3, pp. 13–19, 2013.
- [4] O. Mirzaei and R. M. Akbarzadeh-T, "A novel learning algorithm based on a multi-agent structure for solving multi-mode resource-constrained project scheduling problem," *Journal of Convergence*, vol. 4, no. 1, pp. 47–52, 2013.
- [5] V. V. M. Nhat and N. H. Quoc, "A model of adaptive grouping scheduling in obs core nodes," *Journal of Convergence*, vol. 5, no. 1, pp. 9–13, 2014.
- [6] A. K. Gopalakrishnan, "A subjective job scheduler based on a backpropagation neural network," *Human-Centric Computing and Information Sciences*, vol. 3, article 17, 2013.
- [7] H. Y. Hsueh, C. N. Chen, and K. F. Huang, "Generating metadata from web documents: a systematic approach," *Human-Centric Computing and Information Sciences*, vol. 3, article 7, 2013.
- [8] N. Ibrahim, M. Mohammad, and V. Alagar, "Publishing and discovering context-dependent services," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, 2013.
- [9] H. Bai, A. Wang, Y. Zhao, J. S. Pan, and A. D. Abraham, *Multiple Description Coding: Principles, Algorithms and Systems*, Springer, New York, NY, USA, 2011.
- [10] T. Gan, L. Gan, and K.-K. Ma, "Reducing video-quality fluctuations for streaming scalable video using unequal error protection, retransmission, and interleaving," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 819–832, 2006.
- [11] K. Park and W. Wang, "Afec: an adaptive forward error-correction protocol for end-to-end transport of real-time traffic," in *Proceedings of the IEEE International Conference on Computer Communication and Networks*, pp. 196–205, Lafayette, La, USA, 1997.
- [12] S. Jakubczak and D. Katabi, "A cross-layer design for scalable mobile video," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11)*, pp. 289–300, September 2011.
- [13] S. Aditya and S. Katti, "FlexCast: graceful wireless video streaming," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11)*, pp. 277–288, September 2011.
- [14] J. J. Quinlan, A. H. Zahran, and C. J. Sreenan, "ALD: adaptive layer distribution for scalable video," in *Proceedings of the 4th ACM Multimedia Systems Conference (MMSys '13)*, pp. 202–213, March 2013.
- [15] Z. Li, Y. Huang, G. Liu, F. Wang, Z.-L. Zhang, and Y. Dai, "Cloud transcoder: bridging the format and resolution gap between Internet videos and mobile devices," in *Proceedings of the 22nd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '12)*, pp. 33–38, June 2012.
- [16] A. G. Kunzel, H. Kalva, and B. Furht, "A study of transcoding on cloud environments for video content delivery," in *Proceedings of the ACM Workshop on Mobile Cloud Media Computing (MCMC '10)*, pp. 13–18, October 2010.
- [17] Z. Huang, C. Mei, L. E. Li, and T. Woo, "CloudStream: delivering high-quality streaming videos through a cloud-based SVC proxy," in *Proceedings of the INFOCOM 2011*, pp. 201–205, April 2011.
- [18] S. Ko, S. Park, and H. Han, "Design analysis for real-time video transcoding on cloud systems," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC '13)*, pp. 1610–1615, March 2013.
- [19] B. Shen, S. Lee, and S. Basu, "Caching strategies in transcoding-enabled proxy systems for streaming media distribution networks," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 375–386, 2004.
- [20] S.-P. Chan, C.-W. Kok, and A. K. Wong, "Multimedia streaming gateway with jitter detection," in *Proceedings of the International Conference on Communications (ICC '03)*, pp. 1875–1879, May 2003.
- [21] C.-M. Huang, T.-H. Hsu, and C.-K. Chang, "A proxy-based adaptive flow control scheme for media streaming," in *Proceedings of the ACM Symposium on Applied Computing*, pp. 750–754, New York, NY, USA, March 2002.
- [22] L. Gao, Z.-L. Zhang, and D. Towsley, "Proxy-assisted techniques for delivering continuous multimedia streams," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, pp. 884–894, 2003.
- [23] P. Schojer, L. Böszörményi, H. Hellwagner, B. Penz, and S. Podlipnig, "Architecture of a quality based intelligent proxy (QBIX) for MPEG-4 videos," in *Proceedings of the 12th International Conference on World Wide Web (WWW '03)*, pp. 394–402, Budapest, Hungary, May 2003.
- [24] S.-P. Chan, C.-W. Kok, and A. K. Wong, "Multimedia streaming gateway with jitter detection," *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 585–592, 2005.

- [25] A. Laursen, J. Olkin, and M. Porter, "Oracle media server: providing consumer based interactive access to multimedia data," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, vol. 23, pp. 470–477, May 1994.
- [26] S. Robertson and R. Rivin, *Analog Devices: Designing IPTV Set-Top Boxes Without Getting Boxed*, Analog Devices.
- [27] Telchemy, *Analog Devices: Designing IPTV Set-Top Boxes Without Getting Boxed*, Telchemy.
- [28] S. Acharya and B. Smith, "Middleman: a video caching proxy server," in *Proceedings of NOSSDAV*, 2000.
- [29] K. L. Wu, P. S. Yu, and J. L. Wolf, "Segment-based proxy caching of multimedia streams," in *Proceedings of the 10th international conference on World Wide*, pp. 36–44, New York, NY, USA, 2001.
- [30] O. Verscheure, C. Venkatramani, P. Frossard, and L. Amini, "Joint server scheduling and proxy caching for video delivery," *Computer Communications*, vol. 25, no. 4, pp. 413–423, 2002.
- [31] J. Liu and B. O. Li, "A QoS-based joint scheduling and caching algorithm for multimedia objects," *World Wide Web*, vol. 7, no. 3, pp. 281–296, 2004.
- [32] "Ns-2 network simulator," 2000, <http://www.isi.edu/nsnam/ns/>.
- [33] ISO/IEC JTC 1/SC29/WG 11 and ITU-T SG16 Q.6, "Joint Scalable Video Model (JSVM) Software Manual," Included in JVT-X203. ISO/IEC JTC 1/SC29/WG 11 and ITU-T SG16 Q. 6., 2007.

## Research Article

# Medical Image Segmentation for Mobile Electronic Patient Charts Using Numerical Modeling of IoT

Seung-Hoon Chae,<sup>1</sup> Daesung Moon,<sup>2</sup> Deok Gyu Lee,<sup>3</sup> and Sung Bum Pan<sup>4</sup>

<sup>1</sup> The Research Institute of IT, Chosun University, 309 Pilmun-daero, Dong-gu, Gwangju 501-759, Republic of Korea

<sup>2</sup> Electronics and Telecommunications Research Institute, 161 Gajeong-dong, Yuseong-gu, Daejeon 305-350, Republic of Korea

<sup>3</sup> Department of Information Security, Seowon University, 377-3 Musimseo-ro, Heungdeok-gu, Cheongju-si, Choong-Chung Buk-do 361-742, Republic of Korea

<sup>4</sup> Department of Electronics Engineering, Chosun University, 309 Pilmun-daero, Dong-gu, Gwangju 501-759, Republic of Korea

Correspondence should be addressed to Sung Bum Pan; sbpan@chosun.ac.kr

Received 16 December 2013; Accepted 31 March 2014; Published 12 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Seung-Hoon Chae 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.

Internet of Things (IoT) brings telemedicine a new chance. This enables the specialist to consult the patient's condition despite the fact that they are in different places. Medical image segmentation is needed for analysis, storage, and protection of medical image in telemedicine. Therefore, a variety of methods have been researched for fast and accurate medical image segmentation. Performing segmentation in various organs, the accurate judgment of the region is needed in medical image. However, the removal of region occurs by the lack of information to determine the region in a small region. In this paper, we researched how to reconstruct segmentation region in a small region in order to improve the segmentation results. We generated predicted segmentation of slices using volume data with linear equation and proposed improvement method for small regions using the predicted segmentation. In order to verify the performance of the proposed method, lung region by chest CT images was segmented. As a result of experiments, volume data segmentation accuracy rose from 0.978 to 0.981 and from 0.281 to 0.187 with a standard deviation improvement confirmed.

## 1. Introduction

Telemedicine is defined by the World Health Organization (WHO) as “the practice of medical care using interactive audiovisual and data communications. This includes the delivery of medical care services, diagnosis, consultation, treatment, as well as health education and the transfer of medical data” [1]. In 1906, Wilhelm Einthoven experimented the first telemedicine by transmitting electrocardiogram (ECG) recordings through telephone [2–4]. Since then, telemedicine has become routine practice for specialists to review remote patients' radiology and neurosurgery image [5, 6]. If we use telemedicine, the information of patient's condition is checked using mobile device remotely as shown in Figure 1.

As a new generation information technology, Internet of Things (IoT) brings telemedicine a new chance, which

applies sensors and network to traditional medical devices, and therefore is able to assign the intelligence to them and implement deeper communication and interaction between patients and remote specialists [7–10]. Besides patients' benefit, IoT even helps entire health industry, in which wide scope of medical devices are connected to existing health network, patient crucial life signal is captured by sensors and transmitted to remote medical center, and the doctor is able to remotely monitor patient condition and provide medical suggestion and aiding [11, 12].

By the improvement of the performance of medical imaging equipment, in accordance with the acquisition of high-resolution digital images, computer image analysis is being actively applied in the field of medical diagnosis and treatment. Recently, through various researches, computer-aided diagnosis (CAD) system showed the results that can



FIGURE 1: Example of telemedicine system.

enhance effect of diagnosis and treatment by assisting the specialists [13]. Especially, the field of medical imaging is growing rapidly by new ways to extract or visualize the organ tissue information from diagnostic medical images obtained by a variety of medical imaging equipment such as X-ray, computerized tomography (CT), magnetic resonance imaging (MRI), ultrasound, and Positron emission tomography (PET) [14]. Ritter et al. divided the major issues in the field of medical image processing into image enhancement, image segmentation, image registration, quantification, visualization, and computer-aided detection [15]. Image segmentation of these is important image processing that needs to be ahead of a variety of medical image processing such as image registration, quantification, visualization, and computer-aided detection. Image segmentation is utilized for not only preprocessing stage of other images' processing but also image compression and protection of medical image. Medical images with high-resolution have difficulties in storage and transmission because they have large data size. So, compression of medical images is needed for effective storage and transmission. We have to protect medical images to prevent making bad uses of them. However, region of interest (RoI) of medical image should not be damaged in processing of compression and protection. Medical image segmentation is needed to compress and protect medical images without the damage of RoI. However, image segmentation is difficult for a radiologist to manually segment the large size of data, and because of the similarity of the biological characteristics of human organs, accurate medical image segmentation is not easy. So, in the field of medical image segmentation,

many researchers are studying a variety of ways to obtain fast and accurate automatic segmentation methods for medical images.

Many methods such as threshold method, watershed, region growing, active shape models (ASM), clustering, and level-set method have been researched for medical image segmentation [16–22]. In performing segmentation, accurate judgment is necessary in order to exactly extract the region of interest from medical images in the presence of other organs. For example, if you want to segment the lung region in chest CT images, the bronchi can be a segment that exists within a chest CT image. In case sufficient information is obtained from the region of the lungs and bronchial region, the segmentation can be performed accurately distinguishing the two regions. On the other hand, in case the size of the region, as shown in Figure 2, is small, the region could not be determined as lung region by the lack of information for selecting the lung region.

In this paper, we researched how to improve the performance of exact segmentation of a small region with volume data which is a bunch of medical images. First, we perform initial segmentation. Small regions are damaged or removed in the initial segmentation process. Damaged or removed small regions need reconstruction to improve performance of segmentation. Therefore, we generate predicted segmentation of slices using volume data with linear equation and proposed improvement method for small regions using the predicted segmentation. Using chest CT images among the medical images, we improved the segmentation result and evaluated the performance through the proposed method.

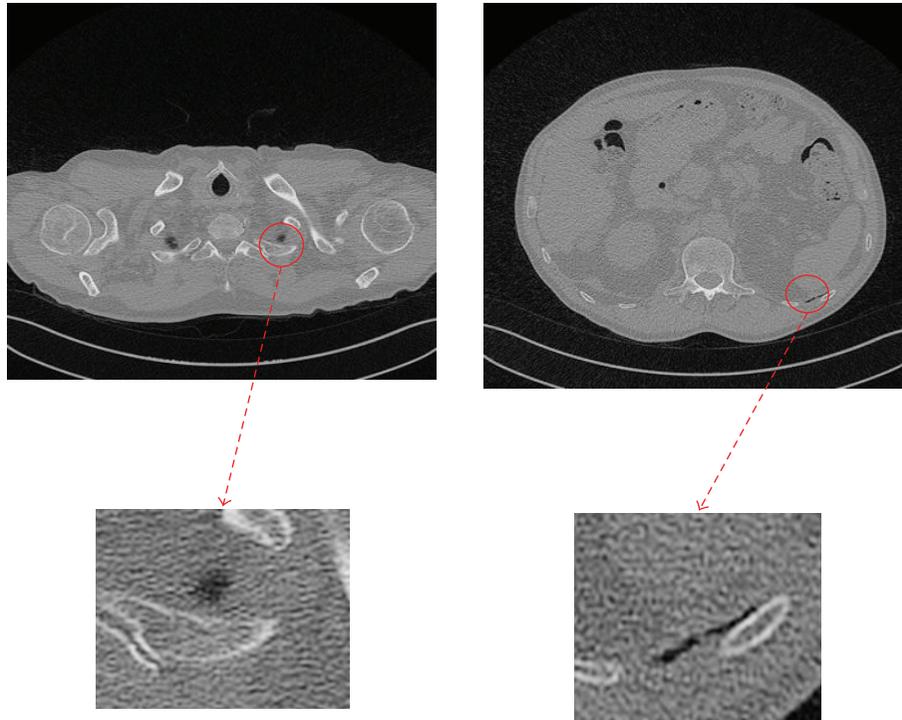


FIGURE 2: The small lung regions.

This paper is organized as follows. In Section 2, proposed improvement method is described. Experiment result is given in Section 3 and conclusion for this paper is made in Section 4.

## 2. Proposed Improvement Method Using Volume Data and Linear Equation

**2.1. Chest CT Image.** The screening of the lungs is important since lung cancer death rate is high among cancers. Among the chest imaging methods, radiograph is a common early screening method which has the advantages of low dose and low cost. In radiograph which expresses a chest on a single image, shadow is generated according to anatomical structures such as ribs and heart. Because radiograph consists of a single image, it is difficult to distinguish pulmonary vascular and lung nodules. By contrast, detection sensitivity of lung nodule using CT is higher than radiograph since CT generates images with volume data. Nation Cancer Institute published the result of research which lowers lung cancer death rate using CT rather than using radiographs in screening [23].

Chest CT images of human body use the 12-bit images instead of general 8-bit images. Generally 300~500 chest CT images region is obtained from a patient, and it varies depending on the performance of the CT scan equipment. Figure 3 shows the representation of three-dimensional modeling of lungs. Because the top and bottom parts of the lungs have diminishing structure becoming smaller and smaller, the lung region of the top and bottom is small. It is difficult

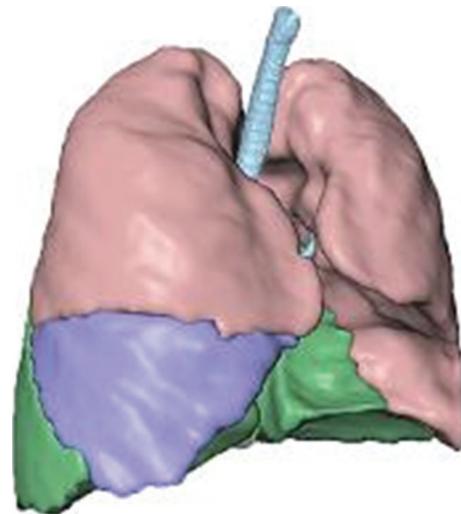
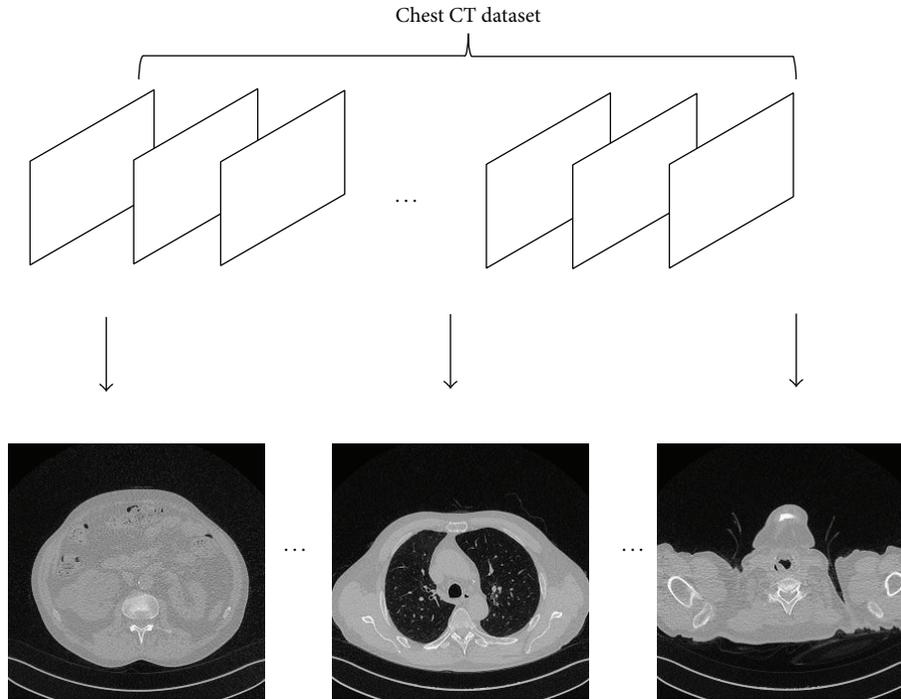


FIGURE 3: Representation of 3D lung modeling.

to determine and segment lung region because the small region of the top and bottom of the lungs does not have many features of the lung.

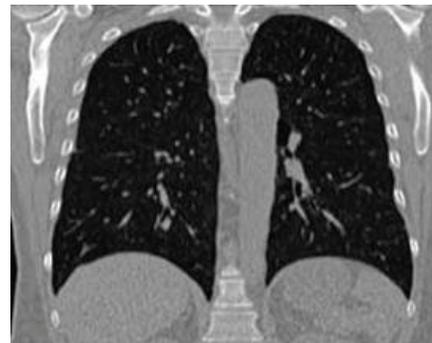
Chest CT images of the dataset consist of axial chest image slices as shown in Figures 4(a) and 4(b). The bundle of axial chest CT images comprising dataset of chest image is able to generate a volume data. Also, it is able to generate the coronal chest image through the volume data as shown in Figure 4(c). Through coronal chest CT image, we can find that shape of lung image does not consist of dramatic changes but



(a) The dataset of chest CT images



(b) Axial chest CT image



(c) Coronal chest CT image

FIGURE 4: The configuration of chest CT dataset.

naturally connected slices. It means that the previous form of lung slices and the next slices are similar to each other and there are natural connections. Using such chest CT images of volume data, predictions of small lung region can reconstruct segmentation results. Accordingly, the performance of lung segmentation was improved.

**2.2. Linear Equation.** The connection form of the coronal plane of the lungs does not have a complex shape. And dramatic changes do not occur because a space between slices is narrow. Therefore, in this paper, segmentation region of next slice is predicted using information of reference slice. To predict region change of next slice, we do not use a method that projects reference slice to next slice but use a method that predicts change of region using linear equation. In addition,

because the information of the next slice is predicted using reference slices, lung region was predicted without using higher-order linear equations, but by the first linear equation which has less computation. The first equation or linear equation is an equation with the highest order term of the order of 1. The first equation may have more than one variable. Linear equations with two variables are actually linear functions as shown in Figure 5. In addition, this is called “equation of the straight line” because it is a straight line in the coordinate plane and associated with the geometric properties of the straight lines.

If given two different points  $(x_1, f_1)$  and  $(x_2, f_2)$ , the equation of a straight line is defined as follows:

$$\begin{aligned} f_1 &= ax_1 + b, \\ f_2 &= ax_2 + b. \end{aligned} \tag{1}$$

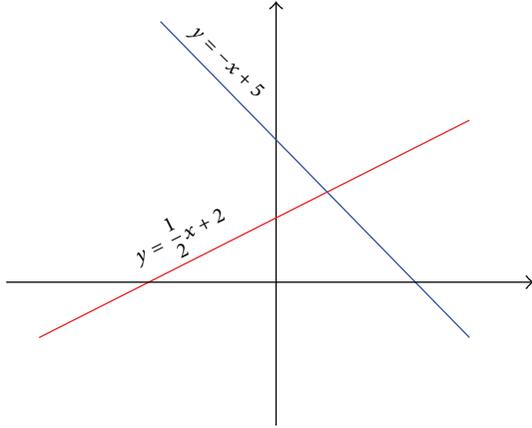


FIGURE 5: The graph of linear equation.

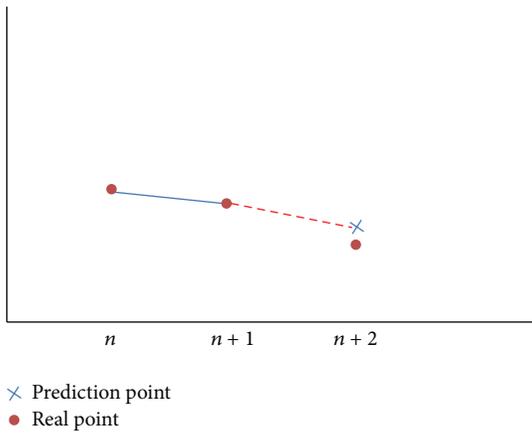


FIGURE 6: Principles of prediction using linear equations.

Using (1), a random coordinate  $(x, f)$  on the line can be obtained as the following equation:

$$\begin{aligned}
 a &= \frac{f_2 - f_1}{x_2 - x_1}, \\
 b &= f_2 - \left( \frac{f_2 - f_1}{x_2 - x_1} \right) x_2, \\
 f &= \left( \frac{f_2 - f_1}{x_2 - x_1} \right) (x - x_1) + f_1.
 \end{aligned}
 \tag{2}$$

That is, given  $(x_1, f_1)$  and  $(x_2, f_2)$ ,  $(x_n, f_n)$  is able to be predicted through the linear equations. As shown in Figure 6, after contour information of lung is extracted in the  $n$ th and  $n + 1$ th slices of chest CT images, the coordinates similar to the actual contours coordinates can be obtained if the outline of the results of the  $n + 2$ th second installment is predicted through the linear equation.

**2.3. Improvement of Segmentation.** Figure 7 is a flow chart of the whole, reconstructing the segmentation results to improve performance of segmentation. First, perform initial segmentation of lung region of each slice of chest CT image

dataset. In order to set the reference slices, measure the dispersedness of each slice and select two consecutive slices with the lowest dispersedness. After selecting anchor points on the contours of the reference slices, adjust results for the segmentation.

In order to apply linear equations to segment region, the coordinate information of contour should be extracted. Using the coordinate information of all contour points is not efficient. So, after setting the anchor point at regular intervals, linear equations between the anchor points should be obtained. As shown in Figure 8(a), set the anchor points on contour of the initial segmentation results of  $n$  and  $n + 1$ th slices at regular intervals. Then, correlation pairs of anchor points are selected using the distance anchor points between  $n$  and  $n + 1$ th slices. Here, the pair of the shortest distance will become a correlation pair of anchor point. Figure 8(b) shows the difference between the initial (dotted line of red) and predicted (line of blue) segmentation results. After searching for a pair of fixed points of the shortest distance between the two results, use the initial segmentation results if they exist within a certain distance. On the other hand, if the initial segmentation results do not exist within a certain distance, adjust segmentation results using the predicted results as the final result.

Lung region can be generated on lung slices in which the region does not exist using a linear equation to predict the contour of the segmentation results. Lung candidate region information was generated by the threshold in order to solve this problem, and the predictions were applied. Using reference image  $m_n$  and  $m_{n+1}$ , generate  $P_{n+2}$  segmentation prediction. Here,  $m_n$  is initial segmentation result of  $n$ th slice. Then, predicted segmentation  $P_{n+2}$  is combined with  $m_{n+2}$  to improve segmentation result of  $m_{n+2}$ . As a result of combining  $P_{n+2}$  and  $m_{n+2}$ , we generate  $M_L$ .  $M_L$  is able to be included artificial regions, since  $P_{n+2}$  is not real result but predicted result. In order to reduce error such as artificial regions, segmentation information  $T_{n+2}$  using threshold was used.  $T_{n+2}$  is initial segmentation information which contains all the regions of lung and bronchi.  $T_{n+2}$  is combined with  $M_L$  using AND operation. Therefore, the errors that generate a lung region in slice which has not real lung region do not occur in final segmentation result  $I_{n+2}$ . This can be expressed as the following equation:

$$\begin{aligned}
 P_{n+2} &= F(m_n, m_{n+1}), \\
 M_L &= P_{n+2} + m_{n+2}, \\
 I_{n+2} &= M_L \cap T_{n+2}.
 \end{aligned}
 \tag{3}$$

In order to obtain such linear equations at least two reference slices are needed. Objects having simpler shapes are the lower probability of segmentation fault. Therefore, we used the dispersedness which can express the simplicity of the form in a numerical value to automatically select the reference slice. By the perimeter of the image ( $p$ ), and the region of the image ( $a$ ), the dispersedness can be summarized

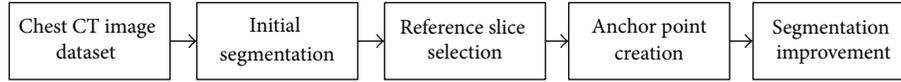


FIGURE 7: The flow chart of proposed segmentation method.

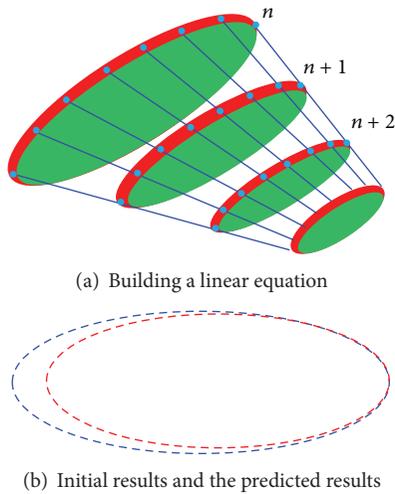


FIGURE 8: Calibration process using a linear equation.

as follows [24]. The lower dispersedness means the object with the simpler shape:

$$D = \frac{p^2}{a}. \quad (4)$$

The results of performing segmentation improvement generate rough segmentation results because interval occurs between anchor points set for the linear equation. Therefore, an extension of the segmentation results fit to the object is necessary. In this paper, the segmentation results using the level-set segmentation method were extended to fit the object.

### 3. Experimental Results

In this paper, to improve the performance of segmentation Vessel Segmentation in the Lung, 2012 (VESSEL12) DB using experiments, was carried out. The VESSEL12 was held as a workshop of International Symposium on Biomedical Imaging 2012 (ISBI2012) introduced through the Grand Challenges in Medical Image Analysis [25]. VESSEL12 DB consists of a total of 20 chest CT image dataset and segmentation mask dataset which is lung region segmentation information. Chest CT images in VESSEL12 dataset are composed of  $512 \times 512 \times 12$  bit images. The segmentation mask file consists of a  $512 \times 512 \times 8$  bit and lung region was classified as 0, 1. One dataset consists of an average of 430 slices and a total of 8,593 chest CT image slices. Dice's overlap methods were used to measure lung region segmentation performance. Result of lung segmentation  $A$  and segmentation mask image  $B$  in

TABLE 1: Experiment result.

|          | Segmentation without proposed method | Segmentation with proposed method |
|----------|--------------------------------------|-----------------------------------|
| S        | 0.978                                | 0.981                             |
| Standard | 0.281                                | 0.187                             |
| Q1       | 0.979                                | 0.982                             |
| Median   | 0.980                                | 0.981                             |

VESSEL12 DB was calculated using the following equation [26]:

$$\text{Score} = \frac{2(A \cap B)}{(A + B)}. \quad (5)$$

Figure 9 shows the appearance that did not segment small lung region and the result of the proposed method reconstructs the segmentation results. Before using the proposed method, small lung region was removed in the segmentation process and lung region determination process. However, the segmentation improvement method using volume data and linear equations shows segmentation result restoring the small lung region as shown in Figure 9(c).

Level-set method was used as the method to initial segmentation for medical images; DRLS was used for the speed function of the level-set method [27]. Table 1 shows the performance of the segmentation method with and without the proposed method. Score for chest CT imaging of a volume data (S) was measured using Dice's overlap, standard deviation of score (Std), and first quartile (Q1), and median for each slice was measured. Q1 is the median of the lower half of the data set. This means that about 25% of the numbers in the data set lie below Q1 and about 75% lie above Q1. Compared to the conventional method, score of the proposed method was improved from 0.978 to 0.981 and the standard deviation was improved from 0.281 to 0.187. Also, we confirmed to improve performance of segmentation of each slice through reduced Q1 and median. Because the size of the improved segmentation region through proposed improvement method was small, the overall accuracy of the impact was small. But, as shown in Figure 9, even in the slice of which lung region is too small to perform lung region segmentation, lung region segmentation was performed.

### 4. Conclusions

As the performance of medical imaging equipment is improving, medical diagnostic using a computer-assisted image analysis is becoming more important. Telemedicine and IoT enable that specialist can consult the patient's condition despite they are in different place. Also, as the specialist uses

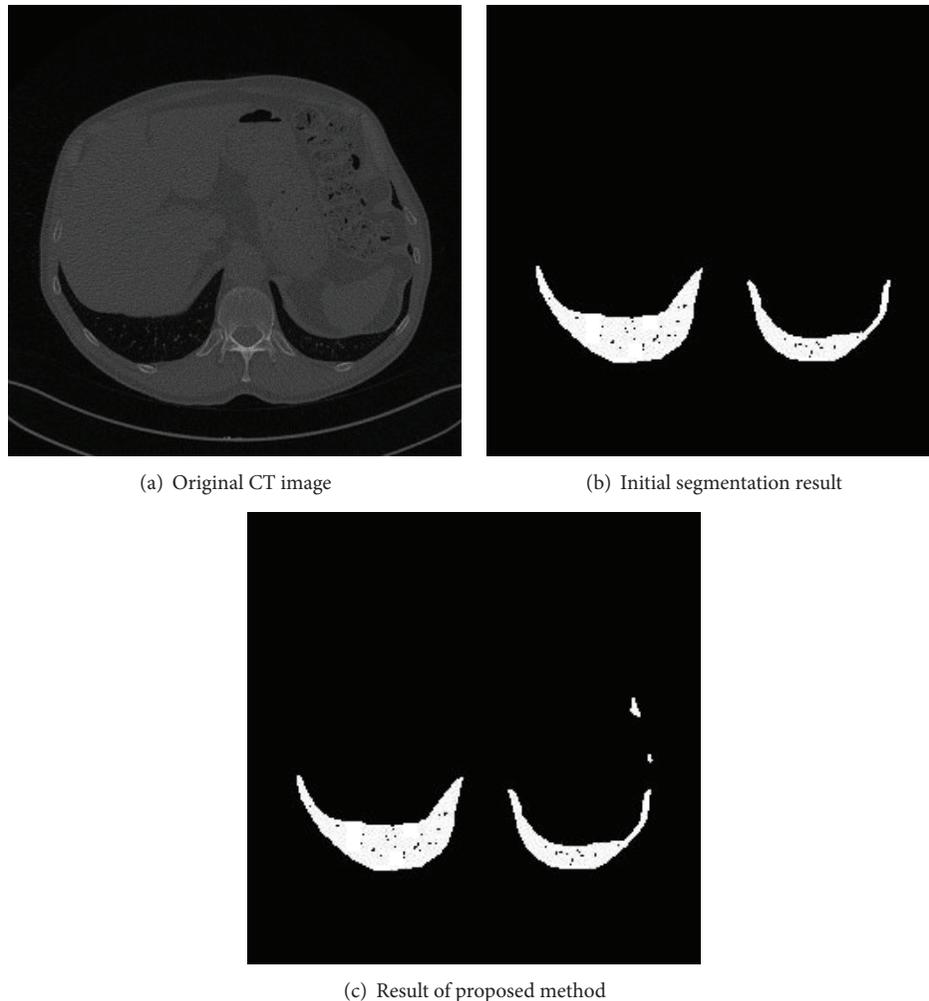


FIGURE 9: Result of lung segmentation improvement.

CAD auxiliary, researches published about what can enhance effect of diagnosis and treatment by using CAD accessorially. In order to effectively use the medical images, many researchers have been researching a variety of methods for fast and accurate segmentation of medical images. In performing segmentation, accurate judgment of region is necessary in order to exactly extract the region of interest from medical images in the presence of other organs. However, the damaged or removed regions occur by the lack of information to determine the interest region in a small region. Damaged or removed small regions need reconstruction to improve performance of segmentation in medical images. Because the top and bottom parts of the lungs have diminishing structure becoming smaller and smaller, the lung region of the top and bottom is small. It is difficult to determine and segment lung region because small region of the top and bottom of the lungs does not have many features of the lung.

In this paper, we researched how to reconstruct the performance of exact segmentation of small region with

volume data and linear equation. The performance of segmentation can be improved through reconstruction of small lung region. Through coronal lung image, we can find that shape of lung image does not consist of dramatic changes but naturally connected slices. Therefore, linear equations using two reference slices can predict the segmentation region of the next slice. Using dispersedness of initial segmentation results, two reference slices were selected, and then anchor points were set on the contour of initial segmentation region in the slices. After obtaining a linear equation using a pair of anchor points in the two slices, segmentation region of the next slice of the reference slice was predicted. By the combination of the predicted results and the initial segmentation result, segmentation of small region was reconstructed. As a result of experiment, we could confirm restructuring damaged or removed small lung regions in chest CT images. And performance of segmentation was improved from 0.978 to 0.981. In particular, the standard deviation of the slices of the volume data is improved 18.7% from 0.281 to 0.187, and

even improvement of segmentation performance in each slice was confirmed.

In the future, we plan to perform image segmentation using a variety of medical imaging DB and conduct researches to detect lesions which exist within the segmented region.

## Conflict of Interests

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

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0023147).

## References

- [1] WHO Website, <http://www.who.int>.
- [2] D. Hailey, R. Roine, and A. Ohinmaa, "Systematic review of evidence for the benefits of telemedicine," *Journal of Telemedicine and Telecare*, vol. 8, no. 1, pp. 1–7, 2002.
- [3] P. T. Moore and M. Sharma, "Enhanced patient management in a hospital setting," *Journal of IT Convergence Practice*, vol. 1, no. 4, pp. 1–23, 2013.
- [4] S. Park, D. K. Shin, and J. S. Kim, "Components of computer-aided diagnosis for breast ultrasound," *Journal of IT Convergence Practice*, vol. 1, no. 4, pp. 50–63, 2013.
- [5] S. Stowe and S. Harding, "Telecare, telehealth and telemedicine," *European Geriatric Medicine*, vol. 1, no. 3, pp. 193–197, 2010.
- [6] M. Brahami, B. Atmani, and N. Matta, "Dynamic knowledge mapping guided by data mining: application on healthcare," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 1–30, 2013.
- [7] H. Kim, Y. Kim, and J. Chang, "A grid-based cloaking area creation scheme for continuous LBS queries in distributed systems," *Journal of Convergence*, vol. 4, article 1, 2013.
- [8] A. Shinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013.
- [9] X. M. Zhang and C. Xu, "A multimedia telemedicine system in internet of things," in *Proceedings of the Computer Science & Information Technology*, vol. 42, pp. 180–187, 2012.
- [10] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [11] W. Shuicai, J. Peijie, Y. Chunlan, L. Haomin, and B. Yanping, "The development of a tele-monitoring system for physiological parameters based on the B/S model," *Computers in Biology and Medicine*, vol. 40, no. 11–12, pp. 883–888, 2010.
- [12] D.-Y. Fei, X. Zhao, C. Boanca et al., "A biomedical sensor system for real-time monitoring of astronauts' physiological parameters during extra-vehicular activities," *Computers in Biology and Medicine*, vol. 40, no. 7, pp. 635–642, 2010.
- [13] J. T. Dobbins III, "Tomosynthesis imaging: at a translational crossroads," *Medical Physics*, vol. 36, no. 6, pp. 1956–1967, 2009.
- [14] L. Costaridou, *Medical Image Analysis Methods*, Taylor & Francis, 2005.
- [15] F. Ritter, T. Boskamp, A. Homeyer et al., "Medical image analysis," *IEEE Pulse*, vol. 2, no. 6, pp. 60–70, 2011.
- [16] Y. Zhu, X. Papademetris, A. J. Sinusas, and J. S. Duncan, "Segmentation of the left ventricle from cardiac MR images using a subject-specific dynamical model," *IEEE Transactions on Medical Imaging*, vol. 29, no. 3, pp. 669–687, 2010.
- [17] H. Badakhshannoory and P. Saeedi, "A model-based validation scheme for organ segmentation in CT scan volumes," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 9, pp. 2681–2693, 2011.
- [18] Z. Liu, H. Wang, and Q. Li, "Tongue tumor detection in medical hyperspectral images," *Sensors*, vol. 12, no. 1, pp. 162–174, 2012.
- [19] J. Zhang, C.-H. Yan, C.-K. Chui, and S.-H. Ong, "Fast segmentation of bone in CT images using 3D adaptive thresholding," *Computers in Biology and Medicine*, vol. 40, no. 2, pp. 231–236, 2010.
- [20] M. C. J. Christ and R. M. S. Parvathi, "Segmentation of medical image using K-Means clustering and marker controlled watershed algorithm," *European Journal of Scientific Research*, vol. 71, no. 2, pp. 190–194, 2012.
- [21] P. Kaur, A. K. Soni, and A. Gosain, "A robust kernelized intuitionistic fuzzy c-means clustering algorithm in segmentation of noisy medical images," *Pattern Recognition Letters*, vol. 34, no. 2, pp. 163–175, 2013.
- [22] S. Chen, T. Kohlberger, and K. J. Kirchberg, "Advanced level set segmentation of the right atrium in MR," in *Medical Imaging: Visualization, Image-Guided Procedures, and Modeling*, vol. 7964 of *Proceedings of SPIE*, February 2011.
- [23] P. B. Bach, J. N. Mirkin, T. K. Oliver et al., "Benefits and harms of CT screening for lung cancer: a systematic review," *The Journal of the American Medical Association*, vol. 307, no. 22, pp. 2418–2429, 2012.
- [24] A. J. Lipton, "Moving target classification and tracking from real-time video," in *Proceedings of the Applications of Computer Vision*, pp. 8–14, 1998.
- [25] VESSEL Segmentation the Lung 2012 (VESSEL12), <http://vessel12.grand-challenge.org>.
- [26] L. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.
- [27] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance regularized level set evolution and its application to image segmentation," *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3243–3254, 2010.

## Research Article

# REST-MapReduce: An Integrated Interface but Differentiated Service

Jong-Hyuk Park,<sup>1</sup> Hwa-Young Jeong,<sup>2</sup> Young-Sik Jeong,<sup>3</sup> and Min Choi<sup>4</sup>

<sup>1</sup> Department of Computer Engineering, Seoul National University of Science and Technology, 232 Gongneung-ro, Nowon-gu, Seoul 139-743, Republic of Korea

<sup>2</sup> Humanitas College, Kyunghee University, No. 26, Kyunghee-daero, Dongdaemun-gu, Seoul 130-701, Republic of Korea

<sup>3</sup> Department of Multimedia Engineering, Dongguk University, 30 Pildong-ro 1 Gil, Jung-gu, Seoul 100-715, Republic of Korea

<sup>4</sup> Department of Information and Communication Engineering, Chungbuk National University, 52 Naesudong-ro, Heungdeok-gu, Chungbuk, Cheongju 361-763, Republic of Korea

Correspondence should be addressed to Min Choi; [miin.chae@gmail.com](mailto:miin.chae@gmail.com)

Received 16 March 2014; Accepted 3 April 2014; Published 11 June 2014

Academic Editor: Laurence T. Yang

Copyright © 2014 Jong-Hyuk Park 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.

With the fast deployment of cloud computing, MapReduce architectures are becoming the major technologies for mobile cloud computing. The concept of MapReduce was first introduced as a novel programming model and implementation for a large set of computing devices. In this research, we propose a novel concept of REST-MapReduce, enabling users to use only the REST interface without using the MapReduce architecture. This approach provides a higher level of abstraction by integration of the two types of access interface, REST API and MapReduce. The motivation of this research stems from the slower response time for accessing simple RDBMS on Hadoop than direct access to RDMBS. This is because there is overhead to job scheduling, initiating, starting, tracking, and management during MapReduce-based parallel execution. Therefore, we provide a good performance for REST Open API service and for MapReduce, respectively. This is very useful for constructing REST Open API services on Hadoop hosting services, for example, Amazon AWS (Macdonald, 2005) or IBM Smart Cloud. For evaluating performance of our REST-MapReduce framework, we conducted experiments with Jersey REST web server and Hadoop. Experimental result shows that our approach outperforms conventional approaches.

## 1. Introduction

With the fast deployment of cloud computing, MapReduce architectures are becoming the major technologies for mobile cloud computing. Nowadays, we are experiencing a major shift from conventional mobile applications to mobile cloud computing. The demand of Open API-based development stems from the increasing use of smartphone applications [1, 2]. Community portal companies are providing Open API service for access to their service. Within a few years, we can expect a major shift from traditional mobile application technology to mobile cloud computing [3]. It improves application performance and efficiency by off-loading complex and time-consuming tasks onto powerful computing platforms. By running only simple tasks on mobile devices, we can achieve a longer battery lifetime and a greater processing efficiency. This off-loading with the use of parallelism is not

only faster but can also be used to solve problems related to large data sets of nonlocal resources. With a set of computers connected on a network, there is a vast pool of CPUs and resources, and you have the ability to access files on a cloud. In this paper, we propose a novel approach that realizes the mobile cloud convergence in a transparent and platform-independent way. Users need not know how their jobs are actually executed in a distributed environment and need not to take into account their mobile platforms are iPhone or Android. All they have to do is to make use of the REST interface, and need not to know the complex distributed computing API such as Hadoop [4].

The research of MapReduce using REST web service interface is underexplored and most research efforts are still at their initial state [5, 6]. MapReduce is a programming model and an associated implementation for processing and generating large data sets. In this work, we propose a concept

of REST-MapReduce enabling users to use only the REST interface without using the MapReduce architecture; it is the MapReduce framework using REST web service Open API interface. We combine MapReduce and REST Open API into an integrated service as REST-MapReduce [7, 8]. This is because of the slower response time for accessing simple RDBMS on Hadoop than direct access to RDBMS [9, 10]. The slow response time stems from the fact that MapReduce was originally designed for analyzing large data, not for simple RDBMS lookup. Such a job, scheduling, initiating, starting, tracking, and management during MapReduce execution, is not a necessary task for REST Open API service execution. To avoid such a problem, REST-MapReduce framework provides an integrated interface with high performance that supports both REST Open API and MapReduce. At the same time, the REST Open API service is provided by a separated architecture for the REST Open API service with a separate architecture. Likewise, we can overcome such a slow response time of simple RDBM invocation on Hadoop by this integrated interface, but differentiated service.

The rest of this paper is organized as follows. Section 2 describes related works. Section 3 explores the architecture of MapReduce computation processes in our REST-MapReduce framework. Section 4 presents the platform independent implementation of application using the REST-MapReduce interface. Section 5 shows performance evaluation. Finally, we conclude and summary our work in Section 6.

## 2. Related Works

Before we go into more detail, we briefly introduce the REST Open API-based mobile application development approaches. To communicate with remote procedure call between client and server, the interface should be defined first. To this end, web service description language (WSDL) and remote procedure call (RPC) were used for the specification. But, these previous approaches are relatively complicated and highly overloaded. Recently, representational state transfer (REST) architecture was first introduced by Fielding. REST web service is becoming popular and explosively used in the field of application development of web and smartphone. Therefore, today's many Internet companies already provide their services by both traditional SOAP-based web service and RESTful web services [11, 12]. The main differences between REST web service and SOAP/WSDL web service are as follows: due to the complicated characteristics of SOAP-based web services, REST web service has not been introduced. REST web service removes the overhead from encoding/decoding of header and body during message transfer. The REST web service enables users and developers to easily use the web services at remote or local sites. We need not add additional communication layer or protocols for REST web service, but we can easily achieve scalability and performance. This research evaluates the performance of mash-up architectures through RESTful Open API web services on smart mobile devices. It provides the analytical experimental results for the performance evaluation of system models. Especially, we try to find an optimal number of

parallel REST web server architectures under certain request arrival rates. We show the performance of the proposed architecture, especially the mean number of requests in the queue and the mean waiting time.

REST web service is a core technology for smartphone application development. This is because REST web service is the most appropriate way for accessing information through the Internet. Usually, a smartphone application needs information from several sources of (one or more) REST web services [13]. So, we need to utilize two or more REST web services composition to realize a target application [14, 15]. In this paper, we propose a server architecture for managing REST web services. This server is for managing web services so as to provide web server maintenance, especially on composition, deployment, and management of REST web services. It enables service developers to conveniently develop, deploy, upload, and run their composed web services with the use of general OOP languages [16].

In 2004, the concept of MapReduce [17] was first introduced as a novel programming model and implementation for a large set of computing devices. Map generates a set of intermediate key/value pairs and reduces merges all intermediate values associated with the same intermediate key, so that programs with this are automatically parallelized and executed on a large cluster of computing devices [18, 19].

Apache Hadoop has become the de facto standard for managing and processing hundreds of terabytes to petabytes of data. It is an open-source Java software framework that supports massive data processing across a cluster of servers. It can run on a single server, or thousands of servers. Hadoop uses a programming model called MapReduce to distribute processing across multiple servers. It also implements a distributed file system called HDFS [20] that stores data across multiple servers. Hadoop monitors the health of servers in the cluster and can recover from the failure of one or more nodes. In this way, Hadoop provides not only increased processing and storage capacity but also high availability. Hadoop [4] is actively used these days by Amazon/A9 [21], Facebook, Google, IBM [22], Joost, New York Times, PowerSet, Yahoo, and so on.

## 3. REST-MapReduce Framework Architecture

This research focuses on designing the concept of MapReduce using the REST Open API interface. This means that both interfaces of REST Open API and MapReduce are integrated into a REST Open API interface. We provide a higher level of abstraction by integrating those two different types of access methods, such as REST Open API and MapReduce. The abstraction by integration provides higher abstraction for both REST API and MapReduce. Users need not to recognize or differentiate how to use those two interfaces, respectively. This is good for user convenience, but it is known to have lower performance when simple RDBMS access occurs on MapReduce servers. This is because MapReduce was originally designed for analyzing large data through parallel execution among multiple cluster nodes. To avoid such an overhead, we proposed a novel architecture as follows.

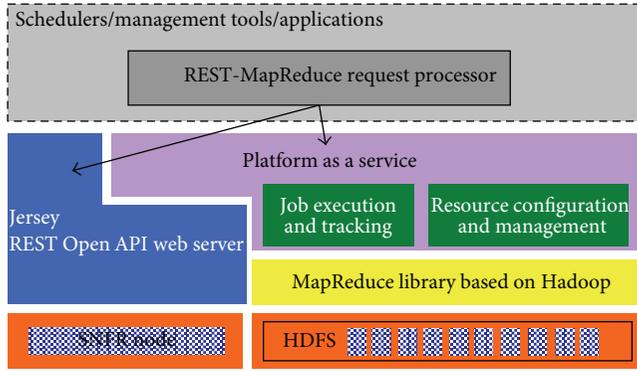


FIGURE 1: REST-MapReduce framework architecture.

3.1. *Architecture.* Figure 1 depicts the architecture of our REST-MapReduce framework. It has five core components: applications, Jersey, platform as service, MapReduce library, and HDFS/S3. First, REST-MapReduce Request Processor acts as the role of a service differentiator in this framework. It determines whether the incoming request is for REST Open API or MapReduce. Then, it sends to Hadoop or Jersey depending on the request type. Second, Jersey is the open source JAX-RS (JSR 311) Reference Implementation [6] for building RESTful Web services. Jersey provides an API so that developers may extend Jersey to suit their needs. We make use of both Tomcat and Jersey to implement our systems. Platform as a Service is achieved by Hadoop. The MapReduce library, job execution, job tracker, and resource management schemes are from the Hadoop. Third, HDFS stands for Hadoop distributed file system, whereas SNFR stands for special node for fast responses [23].

The general concept is that a user submits a job to our REST-MapReduce framework. Then, the REST-MapReduce request processor determines whether the request is for REST Open API or MapReduce. Then, it sends it to either Hadoop or Jersey depending on the request type. Information about the type of the incoming request is necessary for the initial job placement to maximize resource utilization and also that of the entire system. This is because the most appropriate node to execute the task is determined by the type of request. If it is a REST API call, it is better to be forwarded to Jersey server due to its performance, whereas if it is a MapReduce request, it should be forwarded to Hadoop server because of its nature of the parallel execution. The user client can communicate with the PaaS components, such as Resource Configuration & Manager, using the client tool to first acquire a new connection and then submit the application to be run via ClientRMProtocol#submitApplication. As part of the ClientRMProtocol#submitApplication call, the client needs to provide sufficient information to the ResourceManager to “launch” the application’s first container, that is, the ApplicationMaster. You need to provide information such as the details about the local files/jars that need to be available for your application to run, the actual command that needs to be executed (with the necessary command line arguments), any Unix environment settings (optional), and so forth. Effectively, you need to describe the Unix process(es) that

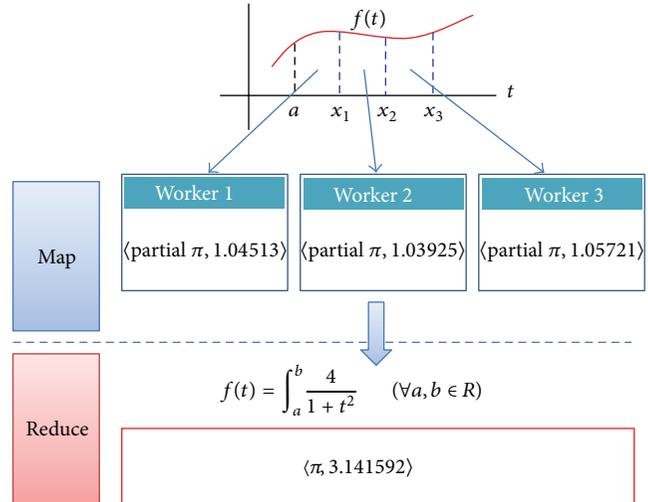


FIGURE 2: MapReduce computation process on our framework.

needs to be launched for your ApplicationMaster. Due to the integration, there are somewhat different features in requests through REST Open API from smartphones. Almost all requests are usually simple data lookup, whereas the rest of them are task/data parallel operations. Therefore, we focus on differentiating those two operations to increase response time.

Figure 2 shows the overall flow of a MapReduce computation process in our REST-MapReduce architecture. When a new job is submitted to a system, a global job scheduler selects the most preferable node for the job to be executed (mapping strategy). Then, the Hadoop JobTracker monitors the job by keeping track of change of job resource usage during the execution. Let us take a look at the procedure in detail. When the user program calls the REST Open API, the following sequence of actions occurs.  $s^*(s^*1)$  The job execution and MapReduce module split the pi value calculation workload into multiple nodes. Then, it starts up on multiple workers of the Hadoop cluster. Our approach is different in terms of task parallelism and not data parallelism. Typically, previous researches in the field of big data processing on Hadoop cluster usually focus on data parallelism, distributing the data of 16 megabytes to 64 megabytes (MB) per piece through the Hadoop cluster.  $s^*(s^*2)$  One of the workers (the workers run on nodes called DataNodes or slaves, interchangeably) has a special purpose. It is a master node. The master node reduces tasks to be assigned. The master picks idle workers and assigns to each one a map task or a reduce task. The rest are slave workers. The slaves are configured in conf/slaves of the Hadoop configuration. They initially join into the framework on system bootup. Once they have joined the framework, the master node sends a short heartbeat message to every worker periodically. If there is no response from a worker within a certain amount of time, the master checks the worker as failed.  $s^*(s^*3)$  After completion of the distributed workload calculation, the Reduce worker iterates over the sorted intermediate data and, for each unique intermediate key encountered, passes the key and the corresponding

set of intermediate values to the user's Reduce function [1]. In this work, we eliminate data dependency through workload parallelization, if any exists, between the workloads of slaves. This is because the data dependency leads to performance degradation, severely resulting in sequential execution.  $s^*$  ( $s^*4$ ) Map phase generates computation result as a form of key-value pairs into log files (e.g., <partial pi>, <1.05721>). The Map function takes a log line, pulls out the timestamp field when the server finished processing the request, converts it into a minute-in-week slot, and then writes out in file systems. Reduce phase reads and sorts all intermediate data so that all occurrences of the same key are grouped together, resulting in the final result which is numerically added for all the same keys. This is the reason why we can see the final pi value as 3.141592 in Figure 2.

**3.2. Task Parallelization Phase.** In this section, we show a development procedure of the cloud-based applications on a mobile platform, especially  $\pi$  calculation. The first step in this procedure is to identify sets of tasks that can run concurrently and/or partitions of data that can be processed concurrently. The second step is to eliminate dependency, if any exists, between every computational phase in the algorithm. The dependency limit of the degree of parallelism results in performance degradation.  $\pi$  is a mathematical constant whose value is the ratio of any Euclidean plane circle's circumference to its diameter; this is the same value as the ratio of a circle's area to the square of its radius. Many formulas from mathematics, science, and engineering involve  $\pi$ , which makes it one of the most important mathematical constants. The simplest method to calculate  $\pi$  is circumference divided by diameter [24]. However, it is difficult to get the exact circumference using this simple method. As a result, there are other formulas to calculate  $\pi$ . These include series, products, geometric constructions, limits, special values, and pi iterations. To calculate  $\pi$  through mobile cloud convergence, we first need to convert the algorithm into a parallelized version. We present a  $\pi$  calculation with infinite series that puts forth a parallelization method for ease of application on the mobile cloud convergence. To calculate  $\pi$ , we first show the procedure of parallelizing the pi calculation as follows:

$$P_n(x) = f(c) + f'(c)(x-c) + \frac{f''(c)}{2!}(x-c)^2 + \dots + \frac{f^{(n)}(c)}{n!}(x-c)^n, \quad (1)$$

where  $P_n(x)$  is defined by the Taylor series.  $P_n(x) = \sum_{k=0}^{\infty} (f^{(k)}(0)/k!)$ , especially on  $c = 0$  is known as the Maclaurin series. So, we compute the Maclaurin series generated by  $f(x) = \tan^{-1}(x)$ . Since we need the  $n$ th order derivative of  $f(x) = \tan^{-1}(x)$ , we apply this expression to the Maclaurin series. Consider

$$P_n(x) = 0 + x + 0 + \frac{x^3}{3} + 0 + \frac{x^5}{5} + \dots = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots \quad (2)$$

Finally, we get the following expression from  $P_n(x) = \tan^{-1}(x)$ :

$$\tan^{-1}(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots \quad (3)$$

But, there is still another problem such that a function to compute this based on the above form is not appropriate for parallelization. This is because each computed value is dependent on previously computed values. Assuming we distribute this workload on eight nodes, they should not be dependent on the previous iteration and the next iteration. That means the next term calculation requires the result of previous term calculation, resulting in serialized execution in a parallelized environment. For example, considering the following expression:

$$\tan^{-1}(x) = \frac{\pi}{4}, \quad (4)$$

it is necessary to calculate the following expression:

$$\tan^{-1}(x) = 1 - \frac{1^3}{3} + \frac{1^5}{5} - \frac{1^7}{7} + \dots \quad (5)$$

But, for computing  $(-1^7/7)$ , the partial term of  $1 - 1^3/3 + 1^5/5$  should be calculated a priori. Again for computing  $(+1^5/5)$ , the partial term of  $1 - 1^3/3$  should be calculated a priori. Thus, we need to come up with a parallelized solution for the  $\pi$  calculation.

In this paper, we propose such a parallelized solution to distribute the heavy workloads to multiple nodes. An independent form of this equation should be provided. Therefore, we convert the equation into an integral form that is suitable for parallelized execution on MapReduce framework. We first take the derivative from the expression (3) with respect to  $x$

$$\frac{d}{dx} \tan^{-1}(x) = 1 - x^2 + x^4 - x^6 + \dots + (-1)^n x^{2n} + \dots \quad (6)$$

We replace the variable  $x$  with  $t$  for the sake of convenience:

$$\frac{d}{dt} \tan^{-1}(t) = 1 - t^2 + t^4 - t^6 + \dots + (-1)^n t^{2n} + \dots \quad (7)$$

At this time, expression (7) can be simplified by

$$\frac{1}{1+t^2} = 1 - t^2 + t^4 - t^6 + \dots + (-1)^n t^{2n} + \dots \quad (8)$$

to

$$\frac{d}{dt} \tan^{-1}(t) = \frac{1}{1+t^2}. \quad (9)$$

Integrate this formula to infinite

$$\int_a^b \frac{t}{dt} \tan^{-1}(t) = \int_a^b \frac{1}{1+t^2} \quad (\forall a, b \in R). \quad (10)$$

Integrating this equation for the interval  $a$  to  $b$  yields the integral form of  $\tan^{-1}(t)$ . By substituting  $\pi/4 = \tan^{-1}(t)$  into

this formula, we get the parallelized form that is executable on the MapReduce platform:

$$\tan^{-1}(t) = \int_a^b \frac{1}{1+t^2} \quad (\forall a, b \in R). \quad (11)$$

By  $\tan^{-1}(\tan(\pi/4)) = \tan^{-1}(1)$ , we get  $\pi/4 = \tan^{-1}(1)$ . Finally, we make use of (11) in this expression to get the following expression:

$$\pi = 4 \tan^{-1}(t) = 4 \int_a^b \frac{1}{1+t^2} = \int_a^b \frac{4}{1+t^2} \quad (\forall a, b \in R). \quad (12)$$

We approximately get the  $\pi$  value by integrating this equation for the interval  $-1/2$  to  $1/2$ .

Unlike an infinite series representation, the integral form is fully parallelizable and it is easy to divide the problem into chunks/parts of the work. We distribute and map these tasks onto multiple clouding nodes. However, this equation cannot be executed on cloud computing which is highly parallelized and distributed in a computing environment. This is an example of task parallelization and partitioning and can be run on a mobile cloud convergence platform.

## 4. REST Application Interface

**4.1. Persistent Storage.** In this section, we examine local storages on HTML5 web applications. Usually, we make use of cookie and session for keeping information on desktop when network connection is not available. HTML5 provides more options than conventional web development methods. LocalStorage, sessionStorage, and WebDB are like that. While all of these functionalities are applicable to both the mobile and desktop worlds, in the world of desktops you generally have a lower rate of adoption. However, any mobile device released in the last 2-3 years will support most of these specs. Moreover, the explosive use of mobile devices such as smartphones requires the demand of using HTML5 due to one source multiuse (OSMU) development. So, there are big demands of persistent storage using HTML5. The persistent storage support was in demand in the world of desktops, but, with the rise of the mobile web and edge connections, support for offline capability has exploded. Everything from offline data storage to the actual application startup is already available and supported on a wide range of mobile platforms. The HTML5 brings us to the three storage mediums: localStorage, sessionStorage, and WebDB. Luckily, the Sencha Touch data package offers awesome wrappers around all three. We can use these persistent storages regardless of the network connection status. SessionStorage is not a persistent storage, meaning it gets wiped whenever the user leaves the page or closes the application. However, in case of one-page web apps where you stay on the same page the entire time, sessionStorage can be a perfect candidate for offline data access, especially in data-sensitive scenarios where you do not want the data persisted on the device after the user is done using the app. SessionStorage is generally limited to 5 MB in size and when that is exceeded, depending on the platform,

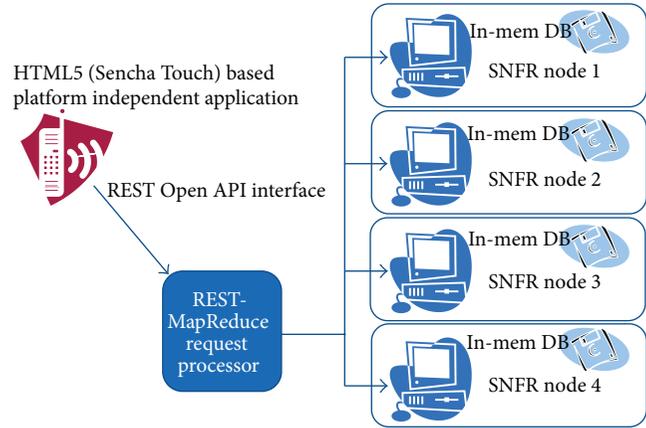


FIGURE 3: System integration and interactions between the components.

either a JavaScript error might be thrown or a popup is presented to the user asking for permission to increase the available storage.

LocalStorage is essentially the same thing as sessionStorage, except that it is persistent. In other words, if you close the app and return, the data will still be there. localStorage is more suited for data that you want to be available when used in combination with the offline startup techniques discussed earlier. However, the localStorage still has a problem for being a perfect persistency. If you clean and delete your web cache from your browser, it will be removed from then on. So, we have to prepare a work around for compensating the cases. Sencha Touch configuration for localStorage looks almost identical to that of sessionStorage.

Finally, the web database is supported by almost every browser. Though specs call for 5 MB limit per origin, iOS has been known to allow up to 50 MB after multiple user prompts. Behind the scenes, it is an SQL database with a query-based language that many of us know and love. When it comes to configuring it in Sencha Touch, it is just as easy as with the other storage mediums.

**4.2. System Components.** We implemented our application exploiting 3 cutting edge technologies. Figure 3 shows the system architecture of our acquisition tax analysis application. Our system architecture consists of the following components: REST Open API server, HTML5 based Platform Independent Client Application, and Database Server. Figure 3 shows the system architecture of our platform independent application design and implementation for checking capital gain tax relief due to one house by one household.

**In-Memory Database.** For the high performance of database, we make use of the in-memory database in this project. Because of too many representative requests, we came up with a state-of-the-art technology for processing this type of short and high frequency requests. The best way to service these requests is the in-memory database.

**REST Open API Web Service.** The REST is a platform independent architectural style. REST ignores the details of

```

Object1 obj1;
String strSearchKeyword = getParameter(STR_PARAM.SEARCHKEYWORD);
String strWebSvcQuery = "http://openapi.naver.com/search?key=test&query=";
strWebSvcQuery += strSearchKeyword + "&target=adult";
URL text = new URL(strWebSvcQuery);
XmlObjectConversionFactory objCreator = XmlObjectConversionFactory.newInstance();
XmlObjectConverter xoConverter = objCreator.newConverter();
obj1 = xoConverter.setInput(text.openStream(), null);
if (obj1.getbAdult()) {
 return;
}
else
{
 try{
 strWebSvcQuery = "http://openapi.naver.com/search?key=test&query=";
 strWebSvcQuery += strSearchKeyword +
 "&display=10&start=1&target=webkr";
 URL text = new URL(strWebSvcQuery);
 String test = text.toString();
 XmlPullParserFactory parserCreator =
 XmlPullParserFactory.newInstance();
 XmlPullParser parser = parserCreator.newPullParser();
 parser.setInput(text.openStream(), null);
 String tag;
 int parserEvent = parser.getEventType();
 while (parserEvent != XmlPullParser.END_DOCUMENT){
 switch(parserEvent){
 case XmlPullParser.TEXT:
 tag = parser.getName();
 break;
 case XmlPullParser.END_TAG:
 tag = parser.getName();
 break;
 case XmlPullParser.START_TAG:
 tag = parser.getName();
 break;
 }
 }
 }catch(Exception e){
 Log.e("dd", "Error in network call"+ e);
 }
}
}

```

ALGORITHM 1: Open API parser.

component implementation and protocol syntax in order to focus on the roles of components, the constraints upon their interaction with other components, and their interpretation of significant data elements.

*Sencha Touch (HTML5) Application.* Sencha Touch is a representative HTML5 UI Framework in these days. Sencha Touch is a well-known user interface (UI) JavaScript library, or framework, specifically built for the Mobile Web. It can be used by Web developers to develop user interfaces for mobile web applications that look and feel like native applications on supported mobile devices. As shown in Algorithm 1, it

is fully based on web standards such as HTML5, CSS3, and JavaScript. Sencha Touch aims to enable developers to quickly and easily create HTML5 based mobile apps that work on Android, iOS, Windows, Tizen, and BlackBerry devices and produce a native-app-like experience inside a browser.

*4.3. System Implementation.* Using these techniques, we developed our system and application which is platform independent one as shown in Figure 3. The application makes use of the AJAX request to the REST Open API web service as shown in Algorithm 2.

```

var button = new Ext.Toolbar({
 cls: "calculator_button",
 height: 35,
 items: [this.text,
 {xtype: 'spacer'},
 {html: new Ext.XTemplate('').apply({name: 'button'}),
 handler: function () {
 Ext.Ajax.request({
 url: '/localhost:8080/Example/apis/example/',
 params: {
 action: 'calculation',
 userid: '15',
 username: 'MCHOI',
 username: 'MCHOI',
 userDate: '20140315',
 },
 success: function(xhr) {
 var response =
 },
 });
 }
]
});
Ext.decode(xhr.responseText);

```

ALGORITHM 2: A portion of our HTML5 application code.

Our application supports WebOS, Android, iOS, Window Phone, and BlackBerry. The application requires the only information of acquisition tax, the area of exclusive space, household numbers, and the location. Then, the application provides the capital gain tax result which is automatically calculated.

## 5. Experimental Results

We describe the experimental result for the REST-MapReduce in this section. This is because REST web service is one of the most convenient methods for accessing information through Internet. Usually, a smartphone application needs information from several sources of (one or more) REST web services. In this experiment, we adopt the Apache Tomcat 7.0 as a web application server, Jersey 1.8 for REST Open API Service Provider, and Hadoop 2.0.4 as MapReduce execution server. Apache Tomcat is an open software with Java Servlet and JavaServer Pages technologies. Apache Tomcat powers numerous large-scale web applications across a diverse range of industries and organizations. Jersey is the open source JAX-RS (JSR 311) Reference Implementation [14] for building RESTful Web services. Jersey provides an API so that developers may extend Jersey to suit their needs. We make use of both

Tomcat and Jersey to implement our systems. We constructed eight-node Linux cluster of Core i5 machines, each with 4 G RAM. The machines were connected by network and managed by Hadoop [4]. Figure 4 shows an overview of our REST-MapReduce framework architecture.

Prior to evaluating the performance in detail, we present system model as a queueing network. The evaluation model of our REST-MapReduce architecture is presented in Figure 4. REST Open API Web Service is composed of three components comprising: (1) dedicated node for Jersey REST web service, (2) Hadoop cluster, and (3) Job schedule/tracker. As shown in Figure 4, there are a number of components (nodes) comprising several queues. Jersey REST web server manages web services instead of web, so as to provide web server maintenance service, especially composition, deployment, and management. Request traverses via the new job submission node and is received by the job scheduler, represented by the components at the left bottom of Figure 4. Our system model is a sort of open queueing network that has external arrivals and departures. The requests enter the system at "New Job Submission" and exit at "OUT" of Hadoop cluster and dedicated node for REST web server, respectively. The number of requests in the system varies with time. In analyzing an open system, we assume that the throughput is known (to be equal to the arrival rate) and we also assume that

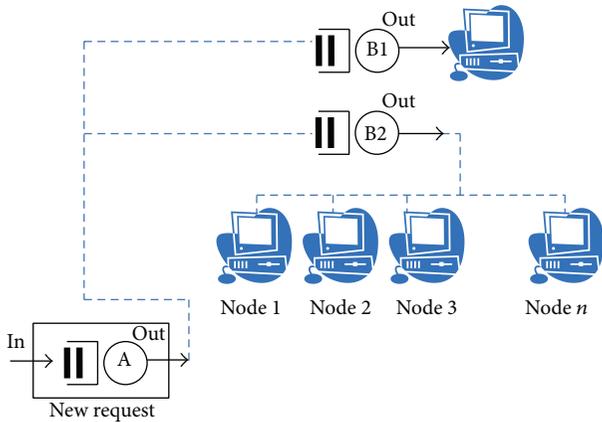


FIGURE 4: Evaluation model.

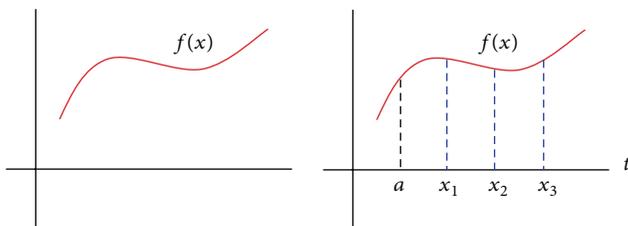


FIGURE 5: Task parallelization and job distribution among Hadoop clusters.

there is no probability of incomplete transfer in this system, so there is no retrial path to go back to Hadoop clusters. The initialization process for the request is done at the scheduler. Then, the job proceeds to the component, either “Hadoop cluster” network or Jersey REST web server, depending on the type of request; if the request is for the REST web server, it goes to the Hadoop cluster. If the request is for just web server, it goes to the web server.

A request may receive service at one or more queues before exiting the system. Jobs departing from the job scheduler arrive at either the Hadoop cluster or dedicated node for Jersey REST web service. All jobs submitted must first pass through the job scheduler/tracker for determining whether it is REST Open API request or MapReduce service. Requests arrive at the web server at an average rate of 1,000/s–15,000/s. Traffic intensity is calculated by the arrival rate over the service rate that means how fast the incoming traffic is serviced on the server.

The key feature of our design is to separate the Jersey web server onto a dedicated node. This feature isolate the performance that is not bound to the MapReduce computation. Hadoop clusters consist of multiple computing nodes. In order to get benefit from such multiple nodes and to handle the heavy load of MapReduce, we need to transform the problem into parallelizable form. To this end, we had the task parallelization phase in Section 3.2. Unlike an infinite series representation, the integral form is fully parallelizable and it is easy to divide the problem into chunks/parts of work. As shown in Figure 5, the total workloads is divided into three

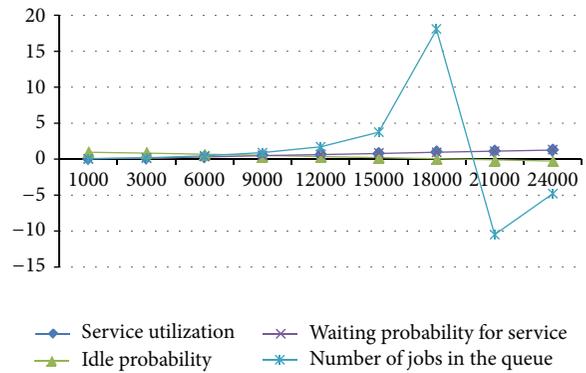


FIGURE 6: Experimental results by increasing service rates 1.

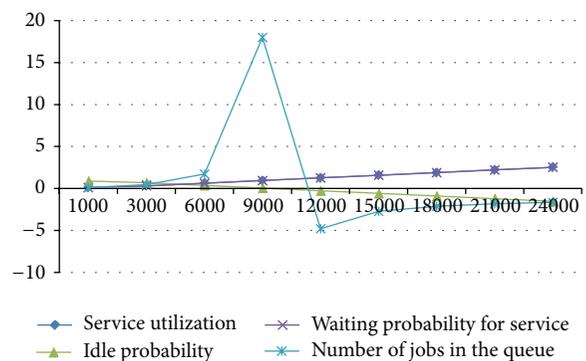


FIGURE 7: Experimental results by increasing service rates 2.

chunks so that we can integrate the formulae at different nodes in parallel. Thus, we can easily distribute and map these tasks onto multiple clouding nodes. We can approximately get the  $\pi$  value by integrating this equation for the interval  $-1/2$  to  $1/2$ .

Figures 6 and 7 show the service utilization, idle probability, waiting probability for service, and number of jobs in the queue depending on increasing service rate. Since the service rate of each Hadoop node in this experiment is 19000 request/sec, the mean number of requests in the queue reaches up to the maximum on the total arrival rate which is increasing between 18000 and 21000. Then, it sharply falls down to the bottom right after the total arrival rate of 21000.

Figure 8 shows the system utilization depending on the change of performance of REST web service. The graph from Va10 to Va300 shows the system utilization by increasing workload on the REST web server. As mentioned above, incoming jobs proceed to the component, either “Hadoop cluster” network or Jersey REST web server, depending on the type of the request; if the request is for the REST web server, it goes to the Hadoop cluster.

If the request is for just web server, it goes to the web server. Thus, if there are large requests incoming for REST web service, then it is natural and there are relatively small requests for MapReduce. This is the reason why the utilization of MapReduce servers gets lower by increasing the server utilization of REST web server.

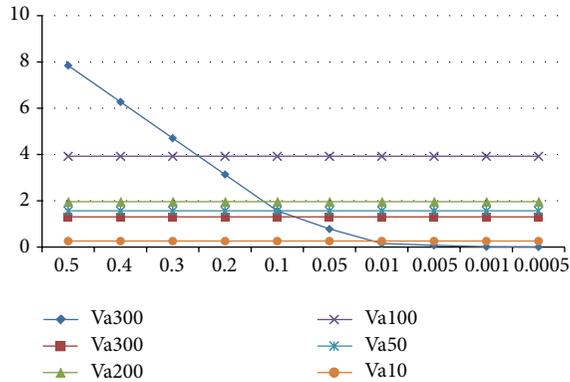


FIGURE 8: System utilization depending on the REST web server performance.

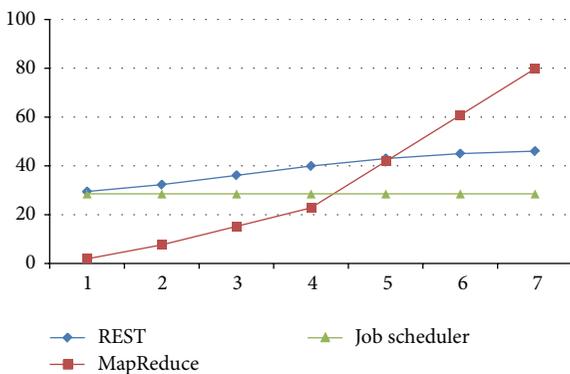


FIGURE 9: Utilization of REST web server, MapReduce clusters, and Job Scheduler.

Figure 9 shows the utilization of REST web server, MapReduce clusters, and Job scheduler. First, the utilization of job scheduler/tracker is constant because the performance change of the job scheduler/tracker is not very high. It is negligible. So, we did not care about the utilization of job scheduler. But we focus onto the utilization of REST web server and MapReduce clusters. By increasing workloads and the number of nodes, the system utilization of MapReduce clusters improves a lot. But, REST web server utilization is just a little bit increased by up to its internal processing limit.

## 6. Conclusion

In this work, we proposed a novel concept of REST-MapReduce, enabling users to use only the REST interface without using the MapReduce architecture. We make both MapReduce and REST Open API into an integrated service as REST-MapReduce. It is well known that there is slower response time for accessing simple RDBMS on Hadoop than direct access to RDBMS. The slow response time stems from the fact that MapReduce was originally designed for analyzing large data, not for simple RDBMS lookup. Such job scheduling, initiating, starting, tracking, and management during MapReduce execution are not necessary tasks for

REST Open API service execution. To avoid such a problem, REST-MapReduce framework provides an integrated interface with high performance that supports both REST Open API and MapReduce. At the same time, the REST Open API service is provided by a separated architecture. Likewise, we can overcome such a slow response time of simple RDBM invocation on Hadoop by this integrated interface, but differentiated service. Surely, we have only focused on the task parallelism such as pi value calculation. But, generally we need to prepare various types of requests for simple DB lookup. So, future work of this research involves trying to make faster DB lookup request on Hadoop framework physically.

## 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 Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0008105).

## References

- [1] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," in *USENIX International Conference on OSDI*, 2004.
- [2] A. C. Murthy, C. Douglas, M. Konar et al., "Architecture of next generation apache hadoop MapReduce framework," Tech. Rep., 2013.
- [3] "Processing and Loading Data from Amazon S3 to the Vertica Analytic Database," White Paper, Amazon Web Service, 2013.
- [4] Hadoop, <http://hadoop.apache.org/>.
- [5] *Amazon Elastic MapReduce Developer Guide*, Amazon Web Service, 2009.
- [6] *Getting Started with Amazon Elastic MapReduce*, Amazon Web Service, March 2009.
- [7] I. Macdonald, "Ruby/Amazon & Amazon web services," *Dr. Dobbs's Journal*, vol. 30, no. 2, pp. 30–34, 2005.
- [8] R. Hussain and H. Oh, "Cooperation-aware VANET clouds: providing secure cloud services to vehicular ad hoc networks," *Journal of Information Processing Systems*, vol. 10, no. 1, pp. 103–118, 2014.
- [9] S. Islam, R. Rahman, A. Roy, I. Islam, and M. R. Amin, "Performance evaluation of finite queue switching under two-dimensional M/G/1(m) traffic," *Journal of Information Processing Systems*, vol. 7, no. 4, pp. 679–690, 2011.
- [10] R. Pan, G. Xu, B. Fu, P. Dolog, Z. Wang, and M. Leginus, "Improving recommendations by the clustering of tag neighbours," *Journal of Convergence*, vol. 3, no. 1, pp. 13–20, 2012.
- [11] H. Zhao and P. Doshi, "Towards automated RESTful Web service composition," in *Proceedings of the IEEE International Conference on Web Services (ICWS '09)*, pp. 189–196, July 2009.
- [12] X. Zhao, E. Liu, G. J. Clapworthy, N. Ye, and Y. Lu, "RESTful web service composition: extracting a process model from linear

- logic theorem proving,” in *Proceedings of the 7th International Conference on Next Generation Web Services Practices (NWeSP '11)*, pp. 398–403, October 2011.
- [13] Z. Li and L. O’Brien, “Towards effort estimation for web service compositions using classification matrix,” 2010.
- [14] C. Pautasso, O. Zimmermann, and F. Leymann, “RESTful web services vs. “Big” web services: making the right architectural decision,” in *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*, pp. 805–814, April 2008.
- [15] R. Alarcon, E. Wilde, and J. Bellido, “Hypermedia-driven RESTful service composition,” *Service-Oriented Computing*, Springer, vol. 6568, pp. 111–120, 2011.
- [16] M. Yoon, Y. K. Kim, and J. W. Jang, “An energy-efficient routing protocol using message success rate in wireless sensor networks,” *Journal of Convergence*, vol. 4, no. 1, 2013.
- [17] “The Internet of Things: In action, The Next Web,” <http://thenextweb.com/insider/2013/05/19/the-internet-of-things-in-action/>.
- [18] J. Rao and X. Su, “A survey of automated Web service composition methods,” in *Proceedings of the 1st International Workshop on Semantic Web Services and Web Process Composition (SWSWPC '04)*, pp. 43–54, July 2004.
- [19] J. Dean and S. Ghemawa, “MapReduce: simplified data processing on large clusters,” in *Proceedings of the 6th USENIX Symposium on Operating System Design and Implementation*, 2004.
- [20] C. Pautasso, “RESTful web service composition with BPEL for REST,” *Data and Knowledge Engineering*, vol. 68, no. 9, pp. 851–866, 2009.
- [21] F. O. Catak and M. E. Balaban, “CloudSVM: training an SVM classifier in cloud computing systems,” in *Pervasive Computing and the Networked World*, pp. 57–68, 2013.
- [22] IBM Smart Cloud, <http://www.ibm.com/cloud-computing/us/en/>.
- [23] M. de Kruijf and K. Sankaralingam, *MapReduce for the Cell B.E. Architecture*, Vertical Research Group. Department of Computer Sciences, University of Wisconsin-Madison, 2010.
- [24] M. Choi, J. Park, and Y.-S. Jeong, “Mobile cloud computing framework for a pervasive and ubiquitous environment,” *The Journal of Supercomputing*, vol. 64, no. 2, pp. 331–356, 2013.

## Research Article

# Estimated Interval-Based Checkpointing (EIC) on Spot Instances in Cloud Computing

Daeyong Jung, JongBeom Lim, Heonchang Yu, and Taeweon Suh

*Department of Computer Science Education, Korea University, Seoul, Republic of Korea*

Correspondence should be addressed to Taeweon Suh; [suhtw@korea.ac.kr](mailto:suhtw@korea.ac.kr)

Received 21 January 2014; Accepted 6 May 2014; Published 28 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Daeyong Jung 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 cloud computing, users can rent computing resources from service providers according to their demand. Spot instances are unreliable resources provided by cloud computing services at low monetary cost. When users perform tasks on spot instances, there is an inevitable risk of failures that causes the delay of task execution time, resulting in a serious deterioration of quality of service (QoS). To deal with the problem on spot instances, we propose an estimated interval-based checkpointing (EIC) using weighted moving average. Our scheme sets the thresholds of price and execution time based on history. Whenever the actual price and the execution time cross over the thresholds, the system saves the state of spot instances. The Bollinger Bands is adopted to inform the ranges of estimated cost and execution time for user's discretion. The simulation results reveal that, compared to the HBC and REC, the EIC reduces the number of checkpoints and the rollback time. Consequently, the task execution time is decreased with EIC by HBC and REC. The EIC also provides the benefit of the cost reduction by HBC and REC, on average. We also found that the actual cost and execution time fall within the estimated ranges suggested by the Bollinger Bands.

## 1. Introduction

Cloud computing is a computing paradigm that constitutes an advanced computing environment that evolved from utility and grid computing. The infrastructure of cloud computing typically includes a collection of interconnected and virtualized computers from parallel and distributed systems. The virtual computers are dynamically provided to consumers as one or more unified computing resources, based on service level agreements (SLA) established through negotiation between the service provider and consumers [1–5]. Typically, cloud computing services provide a high level of scalability of computing resources combined with Internet technology to multiple customers [6]. Currently, there are several commercial cloud systems in service such as Amazon EC2 [7], GoGrid [8], and FlexiScale [9].

In most of these cloud services, there is a notion of an instance to provide users with resources in a cost-efficient manner. An instance means the virtual machine (VM) that serves for the user's need. In general, instances are classified into two types: on-demand instances and spot instances. On-demand instances are charged for the compute capacity on

an hourly basis without the long-term commitment. This frees users from the costs and complexities of planning, purchasing, and maintaining hardware and transforms commonly large fixed costs into much smaller variable costs [7]. On the other hand, spot instances allow users to bid on unused compute capacity and utilize those instances for as long as the current spot price is below their bid. The spot price is changing periodically based on supply and demand. When users' bids meet or exceed the price, they gain access to the available spot instances. If users are flexible as to when applications should run, spot instances can significantly decrease the cost as reported in [7]. Nevertheless, there is a risk of task failures, which occurs when the spot price of the instance becomes higher than the bid price.

To efficiently handle this problem, the checkpointing schemes have been proposed in the research community [10, 11]. The checkpointing saves the execution status of tasks if a certain condition is met and then recovers the task status from the last saved point upon a failure. It allows a reduction in the execution time and cost in an unreliable computing environment. On a legal side, the SLA is typically used for alleviating the uncertainty by specifying service details such

as price and task execution time. SLA specifies the resource allocation and rental terms to consumers in agreement with providers.

In this paper, we propose the estimated interval-based checkpointing (EIC), which improves the efficiency over our previous study [12]. The key idea is adopting the weighted moving average (WMA) and Bollinger Bands. The moving average is a history-based prediction scheme. The WMA sets a different weight for each time interval in the past and calculates the average of the weights. With these weights, the failure occurrence probability is obtained in each interval. The threshold for checkpointing is calculated based on the average failure probability. We apply two thresholds of price and time in EIC. In addition, we use the Bollinger Bands to inform users of estimated execution time and cost. In the stock market, the Bollinger Bands is a well-known analysis method. It is used to measure the high and low value level of the previous trading data. This method is used to predict the price bid in the stock market. We use the Bollinger Bands to calculate both the estimated execution time and the cost.

We have measured the number of checkpoint trials and total cost per spot instance for a user bid. Simulation results show that the EIC outperforms the existing schemes, hour-boundary checkpointing (HBC) [13] and rising edge-driven checkpointing (REC), [11] in terms of the number of checkpoints. Consequently, the EIC minimizes the execution time of applications and the time wasted by task failures.

The rest of this paper is organized as follows. Section 2 briefly describes related work on resource allocation, SLA, fault tolerance, moving average, and Bollinger Bands in cloud computing. Section 3 presents our system architecture. Section 4 presents our SLA, estimation, and checkpoint algorithms based on the price history of spot instances. Section 5 presents performance evaluations with simulations. Finally, Section 6 concludes the paper.

## 2. Related Work

Many researchers and companies have recently studied fault-tolerance techniques in two different environments of cloud computing: reliable environments, with on-demand instances [14, 15], and unreliable environments, with spot instances [11, 13, 16, 17]. The fault-tolerance techniques are more required in unreliable environments. Our study was performed in the latter category of the environments to provide the cost-effectiveness of task execution.

Spot instances are typically used in unreliable environments, and studies on spot instances focus on performing tasks at low monetary costs. The spot instances in the Amazon Elastic Compute Cloud (EC2) offer lower price at the expense of the reduced reliability [18]. Cloud exchange [19] supports the actual price history of EC2 spot instances. In the spot instances environment, there are numerous studies on resource allocation [16, 17], SLA [6, 20, 21], fault tolerance [10, 11, 13, 16], moving average [22, 23], and Bollinger Bands [24, 25].

On the resource allocation side, Voorsluys and Buyya [16] solve the problem of running computation-intensive tasks on

a pool of intermittent VMs. To mitigate potential unavailability periods, the study proposed a multifaceted fault-aware resource provisioning policy. Their solution employs price and runtime estimation mechanisms. The proposed strategy achieves cost savings and stricter adherence to deadlines. Zhang et al. [17] introduced a solution of how best to match customer demand in terms of both supply and price and to maximize the provider's revenue and the customer's satisfaction in terms of VM scheduling. The proposed model is designed to solve the problem of discrete-time optimal control. This model achieves higher revenues than static allocation strategies and minimizes the average request waiting time. Our work differs from [16, 17] in that we focus on reducing the rollback time after a task failure, achieving the cost savings and reducing the total execution time.

On the SLA side, Andrzejak et al. [20] proposed a probabilistic decision model to help users decide a minimum cost according to an SLA between users and Amazon's EC2. The scheme is based on a probabilistic model for the optimization of cost, performance, and reliability. It improves the reliability of service by changing conditions dynamically to satisfy user requirements. Due to the dynamic nature of cloud computing, continuous monitoring of the quality of service (QoS) attributes is necessary to enforce SLAs. Two similar studies [6, 21] focus on cloud resource management in the reliable cloud environment. One is based on SLA monitoring and enforcement in a service-oriented architecture (SOA) [21], whereas the other focuses more on the resource management. The resource manager optimizes a global utility function that integrates both the SLA fulfillment degree and the computational costs [6]. Our paper differs from [6, 21] in that we deal with the resource management in the unreliable cloud environment.

On the fault tolerance side, two similar studies (HBC [13] and REC [11]) proposed enforcing fault tolerance in cloud computing with spot instances. Based on the actual price history of EC2 spot instances, they compared several adaptive checkpointing schemes in terms of monetary costs and job execution time. Goiri et al. [10] evaluated three fault tolerance schemes, checkpointing, migration, and job duplication, assuming that the communication cost is fixed. The migration-based scheme shows a better performance than the checkpointing or the job duplication-based scheme. Voorsluys and Buyya [16] also analyzed and evaluated the impact of checkpointing and migration on fault tolerance using spot instances. Our paper differs from [10, 11, 13, 16] in that we utilize double thresholds for fault tolerance.

On the moving average and Bollinger Bands side, the moving average takes the next observation data using the data in the past [22, 23]. Reference [22] introduced the simple moving average (SMA) and WMA. Reference [23] used the average data to apply weight according to each interval. It evaluates the average of price depending on the weight change. Our paper also adopts WMA to estimate price, execution time, and thresholds based on price history. However, we found that the estimation is not accurate enough. We overcome this shortcoming by applying Bollinger Bands to estimate the execution time and the price ranges. The Bollinger Bands, proposed by Bollinger [24], is a technical

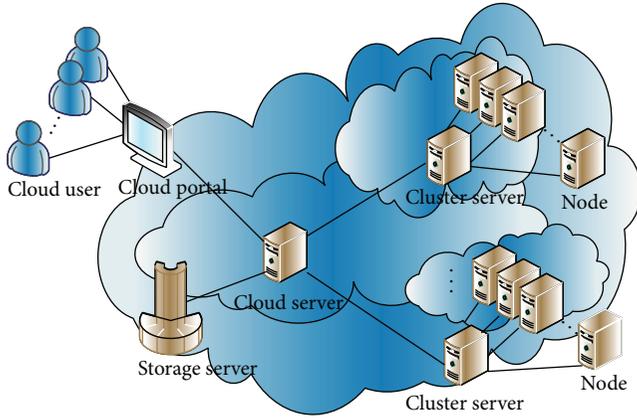


FIGURE 1: Cloud computing environment.

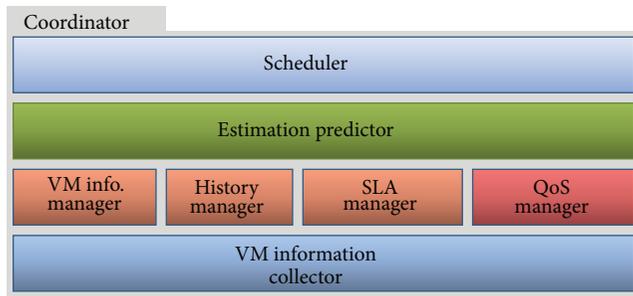


FIGURE 2: Cloud computing environment.

analysis method used in the stock market. It analyzes previous trades and determines the standard deviation. Daytrader [25] introduced a method to predict the range of Bollinger Bands. This prediction requires the selection of length of the moving average around which the Bollinger Bands are plotted, and standard deviations to calculate from this moving average. Our paper differs from [24, 25] in that we apply Bollinger Bands to predict both cost and execution time ranges in the unreliable cloud environment.

In our previous paper [12], we proposed a checkpoint scheme based on SLA to satisfy user requirements. Our previous study performs a checkpointing operation based on two thresholds: price and time. The estimated execution time is predicted using the price history of an instance only for the same amount of time in task execution in the past. This paper differs in that the Bollinger Bands was adopted to improve the accuracy of cost and execution time estimations with utilizing numerous estimation intervals of the past.

### 3. System Architecture

Figure 1 shows the cloud computing environment assumed in this paper, which basically consists of four entities: a cloud server, storage servers, cluster servers, and cloud users. The cloud server is connected to cluster servers and storage servers. The cluster server is composed of many nodes. Cloud users can access the cloud server via the cloud portal to utilize the nodes in the cluster servers as resources. Therefore,

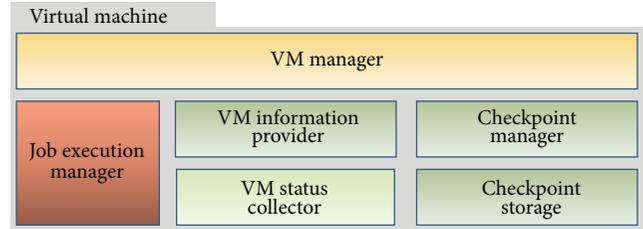


FIGURE 3: The structure of virtual machine.

the cloud server takes responsibility of finding resources and spawning virtual machines to satisfy the user’s requirements in terms of the SLA and QoS. The coordinator in the cloud server manages tasks and is responsible for the SLA management. We focus on the coordinator and the VM, which play important roles in our checkpointing scheme.

**3.1. Layer Structure.** Figure 2 shows the structure of the coordinator in the cloud server, which is composed of Scheduler, Estimation Predictor, VM Information Manager, History Manager, SLA Manager, QoS Manager, and VM Information Collector. In the coordinator, the four managers are responsible for generating and maintaining a list of available VMs, based on the information collected from VM Information Collector. The VM Information Collector collects VM information and provides it to VM information Manager. The VM Information Manager generates a list of CPU utilization, available memory and storage space, network bandwidth, and so on. The History Manager manages the history data, in which the past bid and execution time of spot instances are accumulated. SLA Manager and QoS Manager manage the SLA requirements and the QoS requirements, respectively. Estimation Predictor analyzes data taken from the other managers and calculates the range of estimation completion time and total prices. When a cloud user requests job execution, the Scheduler allocates the requested job to the selected VM.

Figure 3 shows the structure of the VM. In this figure, VM Status Collector collects the status information of the VM, such as CPU utilization and memory space. VM Information Provider extracts resource information needed for job execution using the VM status Collector and delivers the resource information to VM Manager. Job execution Manager executes a requested job received from the coordinator and returns a job result to VM Manager, and VM Manager then delivers the result to the coordinator. Checkpoint Manager manages checkpointing status and the data checkpointed by the Checkpoint Manager are stored in Checkpoint Storage.

**3.2. Instances Types.** The difference between the two instance types is as follows. In on-demand instances, a failure does not occur during task execution, but the cost is comparatively high. In contrast, the cost of spot instances is lower than that of on-demand instances. However, there is an inevitable risk of task failures encountered when the price of the instances becomes higher than the user bid.

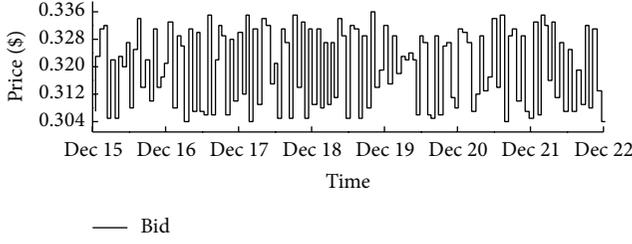


FIGURE 4: Price history of EC2's spot instances for c1-xlarge.

Amazon allows users to bid on unused EC2 capacity among 42 types of spot instances [18]. Their prices, which are referred to as spot prices, are changing dynamically based on supply and demand. Figure 4 shows a spot price fluctuations example during seven days in December 2010 for c1-xlarge (High-CPU Spot Instances—Extra Large) [19]. Our proposed system model is based on the characteristics of Amazon EC2's spot instances.

- (i) The system provides a spot instance when a user bid is higher than the current price.
- (ii) The system immediately stops the spot instance without any notice when a user bid becomes less than or equal to the current price. We refer to it as an out-of-bid event or a failure.
- (iii) The system does not charge for the last partial hour when the system stops the spot instance.
- (iv) The system does charges for the last partial hour when the user voluntarily terminates the spot instance.
- (v) The system provides the spot price history.

#### 4. Estimated Interval-Based Checkpointing

In this section, we detail an estimated interval-based checkpointing for spot instances that includes the SLA, the moving average, Bollinger Bands, and the fault tolerance.

**4.1. Price History-Based SLA.** Figure 5 shows the SLA process between a user and an instance. A user determines an instance type and the bid price to begin tasks on the instance. The coordinator calculates the task execution time based on the user's decision. Then, the coordinator sends a request message to the selected instance to investigate the performance of the instance and calculates the expected execution time, the expected failure time, and the expected cost. Then, the coordinator sends a user the expected execution time and cost. When a task is completed on the selected instance, the coordinator receives the outcome from the instance and sends it to the user. As shown in Figure 5, the prediction function in the coordinator plays an important role in our SLA process because it performs the estimation using price history.

**4.2. Estimation Using Moving Average and Bollinger Bands.** In EIC, the checkpointing operation is performed by analyzing

the price variation at certain time intervals in the past. We use the moving average which estimates a job execution time and a cost from the analyzed data. The estimations are combined with the failure probability to calculate the thresholds for the checkpointing operation. The proper estimation of the execution time and cost is crucial for the credibility of service providers to customers. For the probable estimation information, we use Bollinger Bands. It suggests the upper and lower bounds of the execution time and the cost. We show in Section 5 that the actual execution time and the cost fall within the bounds.

In this paper, we introduce a terminology referred to as estimated interval (EI). Figure 6 shows an illustrative definition of the EI. The detailed definitions are as follows.

- (i) Pure task time: the time to execute a task on a selected instance when there are no failures.
- (ii) Past pure task time: a sum of time durations taken for task execution on the selected instance in the past, excluding failure durations. It is extracted from the price history.
- (iii) Past failure time: a sum of failure durations in the past to execute a task. A failure occurs when the current user bid is below the past spot price.
- (iv) Estimated interval (EI): the sum of the past pure task time and the past failure time.
- (v) Moving average EI: the average of EIs computed using moving average.
- (vi) Expected cost: the average of costs charged for task execution in EIs.

Based on the simple moving average (SMA), we calculate an estimated time  $SMA_{ET}$  and an estimated price  $SMA_{EP}$  by the average of EIs in the price history, as shown below:

$$SMA_{ET}(n) = \frac{ET_1 + ET_2 + ET_3 + \dots + ET_n}{n}, \quad (1)$$

$$SMA_{EP}(n) = \frac{EP_1 + EP_2 + EP_3 + \dots + EP_n}{n},$$

where  $ET_i$  is the estimated time in an interval  $i$ ,  $EP_i$  is the estimated price in an interval  $i$ , and  $n$  is the number of intervals, as depicted in Figure 6.

Based on the weighted moving average (WMA),  $WMA_{ET}$  and  $WMA_{EP}$  are averages of the estimated time and the estimated price from the price history with a weight using SMA. They are calculated by

$$WMA_{ET} = \frac{\sum_{i=1}^n \alpha_i ET_{n-i+1}}{\sum_{i=1}^n \alpha_i}, \quad WMA_{EP} = \frac{\sum_{i=1}^n \alpha_i EP_{n-i+1}}{\sum_{i=1}^n \alpha_i}, \quad (2)$$

where  $\alpha$  is a weight. The  $\alpha_i$  is assigned the highest for the most recent  $EI_1$ , and it is decreased from the most recent  $EI_1$  to the last  $EI_N$ . The weight  $\alpha_i$  is calculated by

$$\alpha_i = \frac{n+1-i}{\sum_{i=1}^n i}, \quad (3)$$

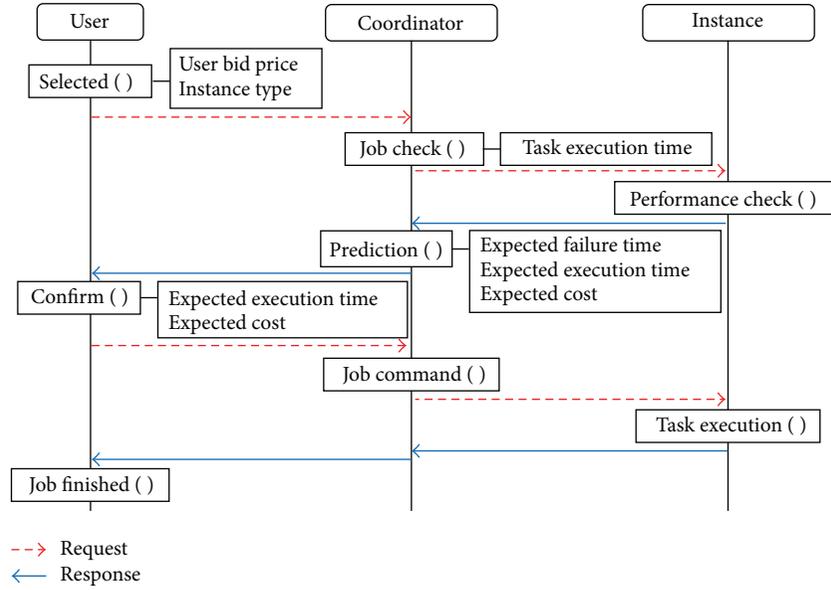


FIGURE 5: SLA processing.

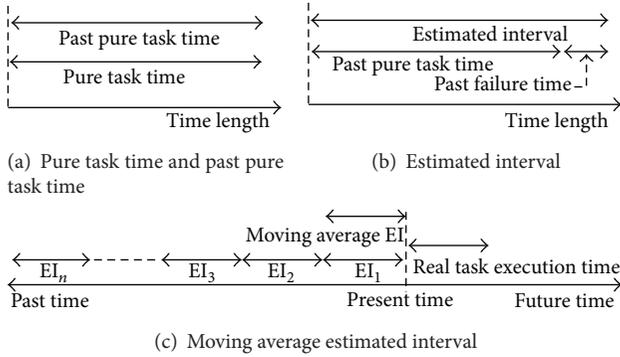


FIGURE 6: Moving average relation.

where  $i$  and  $n$  are the interval number and the last interval number, respectively. By adjusting the weight, we empirically reduce the gap between the estimation and actual data from real execution. The Bollinger Bands presents the range of estimation using a moving average and a standard deviation. Generally, the Bollinger Bands itself adopts a moving average as the middle value. We use WMA as the middle value of the Bollinger Bands because the near past is more likely to be influencing the near future. The upper and lower bounds of the Bollinger Bands are defined as

- (i) Middle Bollinger Band = WMA
- (ii) Lower Bollinger Band = Middle Bollinger Band -  $2\sigma$
- (iii) Upper Bollinger Band = Middle Bollinger Band +  $2\sigma$

where  $\sigma$  is the standard deviation of EIs. Figure 7 illustrates the range of Bollinger Bands using training data that consist of each estimation value in EIs. The training data are obtained from (an)  $N$ -zone EIs.

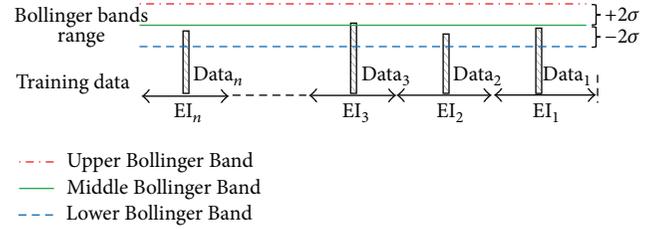


FIGURE 7: Bollinger Bands acquisition.

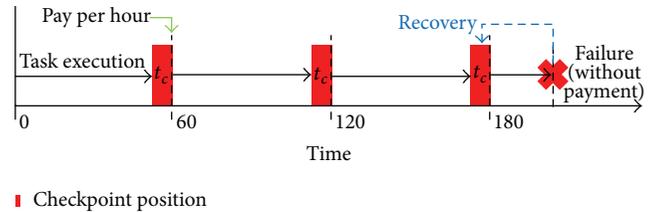


FIGURE 8: Hour-boundary checkpointing.

**4.3. Fault Tolerance Mechanisms Using Checkpoints.** On a spot instance, a task failure occurs when the user's bid is below the current spot price. This problem has been solved by using the checkpointing, one of fault tolerance mechanism [9]. In this section, we detail the existing checkpointing methods and our proposed scheme.

Figure 8 illustrates the hour-boundary checkpointing (HBC). In this scheme, the checkpointing operation is performed in the hour boundary, and a user pays the bidding price on an hourly basis. Upon the task failure, the task is restarted from the position of the last checkpoint.

Figure 9 illustrates the rising edge-driven checkpointing (REC). In this scheme, the checkpointing operation is performed when both the price of the spot instance is raised (i.e.,

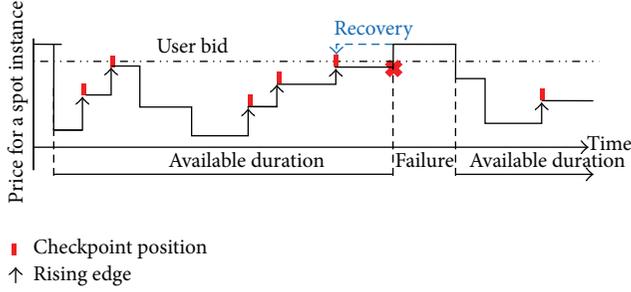


FIGURE 9: Rising edge-driven checkpointing.

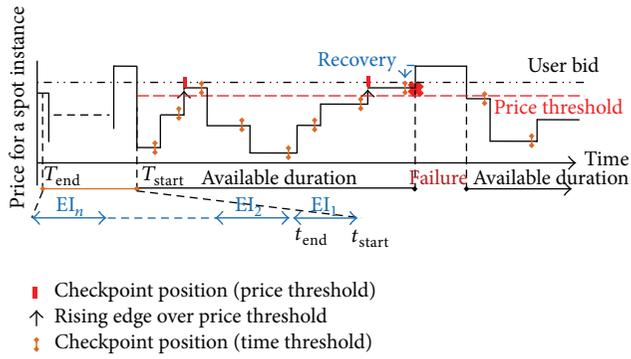


FIGURE 10: Estimated interval-based checkpointing.

rising-edge) and the price is less than the user bid. It increases the number of checkpoints when the spot price fluctuates frequently. The critical problem in REC is that the rollback time becomes long when a rising edge does not appear for a long period of time after a checkpoint is taken. This could lead to a longer time for the task completion than HBC.

Figure 10 illustrates checkpointing operation in EIC. It is basically performed using two thresholds, price and time, based on the expected execution time according to the price history. Now, let  $t_{start}$  and  $t_{end}$  denote a start point and an end point, respectively, in the total of EIs. Based on  $t_{start}$  and  $t_{end}$ , we obtain the price threshold (PriceTh) and the time threshold (TimeTh $_{P_i}$ ), which are used as thresholds in EIC.

The price threshold, PriceTh, can be calculated by

$$\text{PriceTh} = \frac{\text{WP}_{\min} + \text{User}_{\text{bid}}}{2}, \quad (4)$$

where  $\text{User}_{\text{bid}}$  represents the user bid and  $\text{WP}_{\min}$  represents an available minimum price using a moving average in the time duration between  $t_{start}$  and  $t_{end}$ .

First, the  $P_{\min}^{\text{EI}_i}$  represents the minimum price in the time duration between  $T_{start}$  and  $T_{end}$  in  $\text{EI}_i$

$$P_{\min}^{\text{EI}_i} = \text{PriceMin}(t_{start}, t_{end}). \quad (5)$$

Second, the  $\text{WP}_{\min}$  is the average of the product of  $P_{\min}^{\text{EI}_i}$  and sum of the weighted value  $\alpha_i$ :

$$\text{WP}_{\min} = \frac{\sum_{i=1}^n \alpha_i \times P_{\min}^{\text{EI}_{n-i+1}}}{\sum_{i=1}^n \alpha_i}. \quad (6)$$

The time threshold of price  $P_i$ ,  $\text{TimeTh}_{P_i}$ , is calculated by

$$\text{TimeTh}_{P_i} = \frac{\sum_{j=1}^n \alpha_j \times \text{TimeTh}_{P_i}^{\text{EI}_{n-j+1}}}{\sum_{j=1}^n \alpha_j}. \quad (7)$$

In each EI, the time threshold of price  $P_i$ ,  $\text{TimeTh}_{P_i}^{\text{EI}_j}$ , is calculated by

$$\text{TimeTh}_{P_i}^{\text{EI}_j} = \text{AvgTime}_{P_i}^{\text{EI}_j}(t_{start}, t_{end}) \times (1 - F_{P_i}^{\text{EI}_j}), \quad (8)$$

where  $F_{P_i}^{\text{EI}_j}$  is the failure probability of price  $P_i$  and  $\text{AvgTime}_{P_i}^{\text{EI}_j}(t_{start}, t_{end})$  represents the average execution time of  $P_i$  in an interval between  $T_{start}$  and  $T_{end}$  in  $\text{EI}_j$ . The failure occurrence probability  $F_{P_i}^{\text{EI}_j}$  is calculated by

$$F_{P_i}^{\text{EI}_j} = \frac{\sum_{\text{ext}_k \in \text{EI}_j, P_i} \text{ext}_k^{\text{after failure}}}{\sum_{\text{ext}_k \in \text{EI}_j, P_i} (\text{ext}_k^{\text{after failure}} + \text{ext}_k^{\text{after non-failure}})}, \quad (9)$$

where  $\text{ext}_k$  is the execution task time to invoke an interval  $k$  when a price  $p_i$  is in  $\text{EI}_j$ . The after failure function is calculated when the current spot price is above or equal to the user bid. The after non-failure function is calculated when the current spot price is below the user bid.

Using these two thresholds, our scheme performs checkpointing operations in two cases. First, a checkpointing is performed when there is a rising edge in the actual price variation, and the actual price falls in between the user bid and the price threshold; second, the checkpointing is based on the time threshold, which is computed with the failure probability and the average execution time as in (7). It is performed if the current execution time exceeds the time threshold computed at the same past price as the current one.

Algorithm 1 shows the checkpointing and recovery algorithm used in EIC. The flag represents the occurrence of a task failure, and it is initially set to false. The checkpointing process repeats until all tasks are completed. When the task execution is normal (i.e., the flag is false), the scheduler performs a checkpoint operation to cope with the potential job failure (lines 5–25). The scheduler estimates the execution time before the initial task starts (lines 6–9). The recovery process is performed when the flag is true (lines 11–14). The checkpoints are performed in two cases (lines 15–20). If the rising spot price falls in between the user bid and the price threshold, the scheduler performs a checkpointing operation (lines 15–17). If the execution time exceeds the time threshold, the scheduler also performs a checkpointing operation (lines 18–12). When a task failure occurs, the flag is set to true (lines 22–24). Lines 26–29 show the detailed process of time estimation. Lines 30–33 and 34–37 show the detailed process of checkpointing and recovery, respectively.

```

(1) // Input: user's requested task and bid
(2) // Output: total task execution time and total cost
(3) Boolean F_flag = false // a flag representing occurrence of a task failure
(4) Boolean EI_flag = true // a EI_flag representing a task start
(5) while (! Task execution finishes) do
(6) if (EI_flag) then
(7) Estimation();
(8) EI_flag = false;
(9) end if
(10) if (spot prices < User bid) then
(11) if (F_flag) then
(12) Recovery();
(13) F_flag = false;
(14) end if
(15) if (rising edge && Price threshold < spot prices) then
(16) Checkpoint();
(17) end if
(18) if (Time threshold < execution time in current price) then
(19) Checkpoint();
(20) end if
(21) end if
(22) if (failure is occurred) then
(23) F_flag = true;
(24) end if
(25) end while
(26) Function Estimation()
(27) calculate the points of checkpoint to base a price history;
(28) set the price and time thresholds;
(29) end Function
(30) Function Checkpoint()
(31) task a checkpoint on the spot instance;
(32) Send the checkpoint to the storage;
(33) end Function
(34) Function Recovery()
(35) retrieve the checkpoint information form the storage;
(36) restart the job execution;
(37) end Function

```

ALGORITHM 1: Checkpointing and recovery algorithm.

## 5. Performance Evaluation

Our simulations were conducted using the history data obtained from the Amazon EC2's spot instances [19], which was accumulated during a period from December 15, 2010 to December 22, 2010 as shown in Figure 4. The history data before December 20, 2010 were used to extract the expected execution time and failure probability for the proposed checkpointing scheme. The applicability of EIC was tested using the history data after December 20, 2010, which was also used in HBC and REC.

In the simulations, one type of spot instances, *c1.xlarge*, was applied to show the effect of the three checkpointing schemes on performance according to the user bid and the task time. Table 1 shows the applied resource type details used in Amazon EC2. The high-CPU instance offers more compute units than other resources (standard and high-memory instances) and is ideal for the compute-intensive applications. Under the simulation environments, we compare the performance of EIC with those of HBC and REC in

TABLE 1: Resource type information.

| Instance type                  | Compute unit | Virtual cores        | Memory | Storage |
|--------------------------------|--------------|----------------------|--------|---------|
| <i>C1.xlarge</i><br>(High-CPU) | 20 EC2       | 8 cores<br>(2.5 EC2) | 7 GB   | 1690 GB |

terms of various metrics according to the user bid and task time.

*5.1. User Bid Impact on Performance.* Before analyzing the performance of EIC, we extracted the simulation specifics from the spot history presented in Figure 4. Table 2 shows the data used for simulation. The simulations were conducted with incrementing the user bid interval from minimum bid to maximum bid.

We also extracted the failure probability with the current bid price according to each spot price in the past (12-15-2010–12-19-2010), as drawn in Figure 4. The probability was

TABLE 2: Simulation parameters and values.

| Parameter         | Value       |
|-------------------|-------------|
| Task time         | 259,200 sec |
| Checkpoint time   | 300 sec     |
| Recovery time     | 300 sec     |
| Minimum user bid  | \$0.310     |
| Maximum user bid  | \$0.340     |
| User bid interval | \$0.005     |

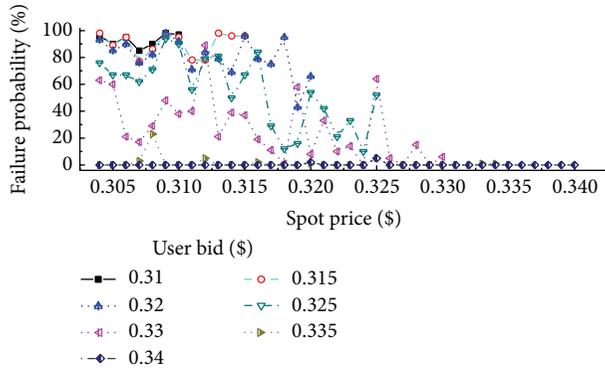


FIGURE 11: Failure occurrence probability.

used to determine the time threshold in EIC. Figure 11 shows the failure occurrence probability for the c1.xlarge instance. The X-axis and Y-axis denote the spot price and the failure probability for a given user bid, respectively.

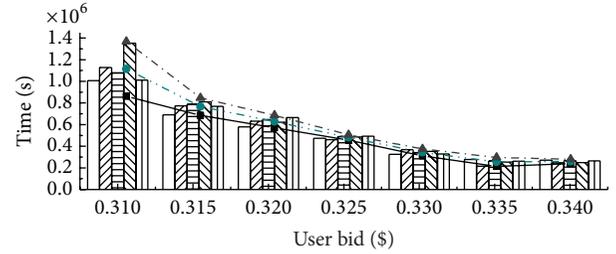
Figure 11 states that the failure occurrence probability changes according to the user bid. As anticipated, if the bid price is low, the failure probability is high across all spot prices. If the bid price is high, the failure probability is low. Thus, it is reasonable to predict that the task execution time will be longer if the failure probability is high because both the total failure time and total rollback time increase.

Figure 12 estimated execution time, cost, and Bollinger Bands of each EI zone computed with the past price history. Figures 12(a) and 12(b) show the execution time and the cost according to the user bid respectively. Estimated interval (EI) with the weighted moving average which is calculated by using the past spot price history, is necessary for the user bid. Figure 12 also shows the Bollinger Bands (Lower\_BB, Middle\_BB, and Upper\_BB) according to the user bid.

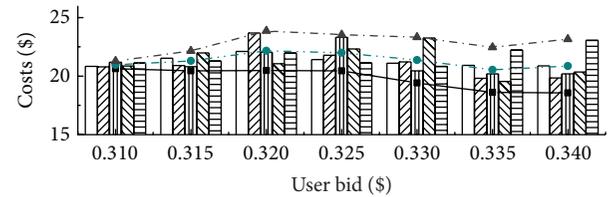
Figure 13 shows the task execution time, the cost, and Bollinger Bands when the number of EI zones is increased. For example, 2 in  $x$ -axis means that two zones (EI<sub>1</sub> and EI<sub>2</sub>) are included in the simulation.

Figure 14 shows the rollback times of EIC, HBC, and REC. The rollback time is calculated from a failure point in time to the last checkpointed time. The EIC lessens the average rollback time by 72.46% over HBC and 88.49% over REC.

Figure 15 shows the performance comparison of EIC, HBC, and REC. The EIC reduces the number of checkpoints on average by 35.97% and 37.92%, compared to HBC and REC, respectively. Consequently, the EIC shortens the task



(a) Estimated Bollinger Bands of execution time



(b) Estimated Bollinger Bands of costs

FIGURE 12: Estimated execution time, cost, and Bollinger Bands of each EI zone computed with the past price history.

execution time by 35.53% over the HBC and 40.40% over REC.

Figure 16 shows the total costs according to the user bid. The EIC reduces the cost on average by 36.26% and 38.52% over HBC and REC, respectively.

Figure 17 shows the combined performance metric, the product of the total execution time, and cost. According to the user bid, the EIC shows marginal variation due to the lowest amount of rollback time among the compared schemes. The EIC achieves the relative benefits in the combined metric on average by 55.73% and 60.95% when compared to HBC and REC, respectively.

Figure 18 shows how well the actual execution time and cost are predicted with EIC according to the user bid. The actual execution time and cost are located between the lower and upper bounds of the Bollinger Band. Figures 18(a) and 18(b) show that they are close to the middle point of the Bollinger Band. The experiments show that the adoption of the Bollinger Band would provide reliable estimations to Cloud users.

**5.2. Task Time Impact on Performance.** In this section, we analyzed the performance of computing-type instances according to the task time. Table 3 shows the simulation parameters. Note that the execution time in simulations

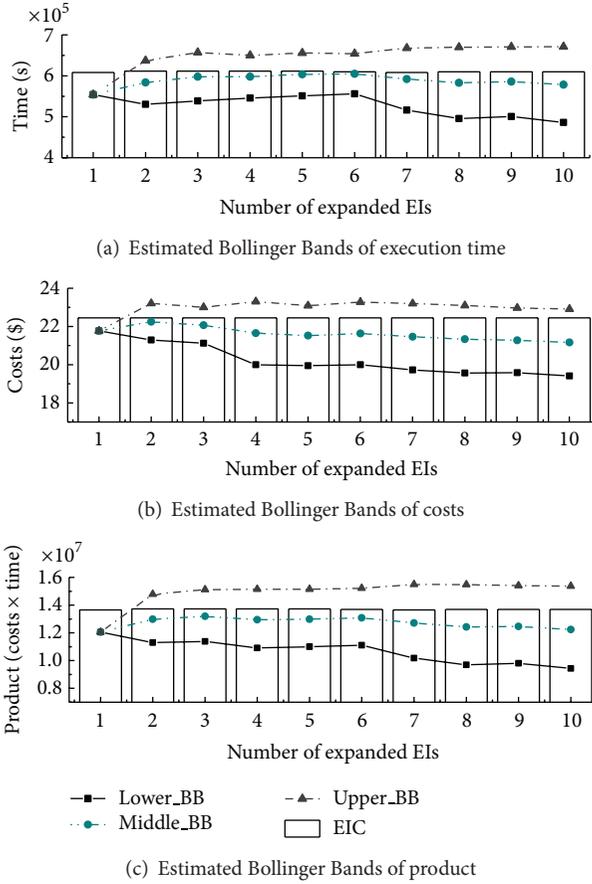


FIGURE 13: Estimated execution time, cost, and Bollinger Bands of expanded EI zones computed with the past price history.

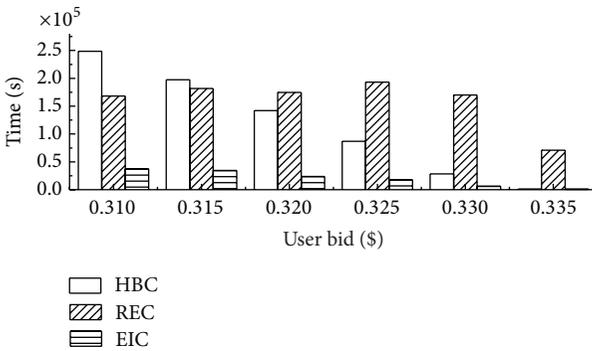


FIGURE 14: Comparison of rollback time according to user bid.

varies from minimum time to maximum time at the granularity of the time interval.

Figure 19 shows the rollback time of EIC, HBC, and REC according to the task time. The increase rate of rollback time in EIC is small compared to HBC and REC. The rollback times are increased by 5.25 times, 15.84 times, and 12.41 times for EIC, HBC, and REC, respectively, when the task times are increased from the minimum to the maximum times. EIC

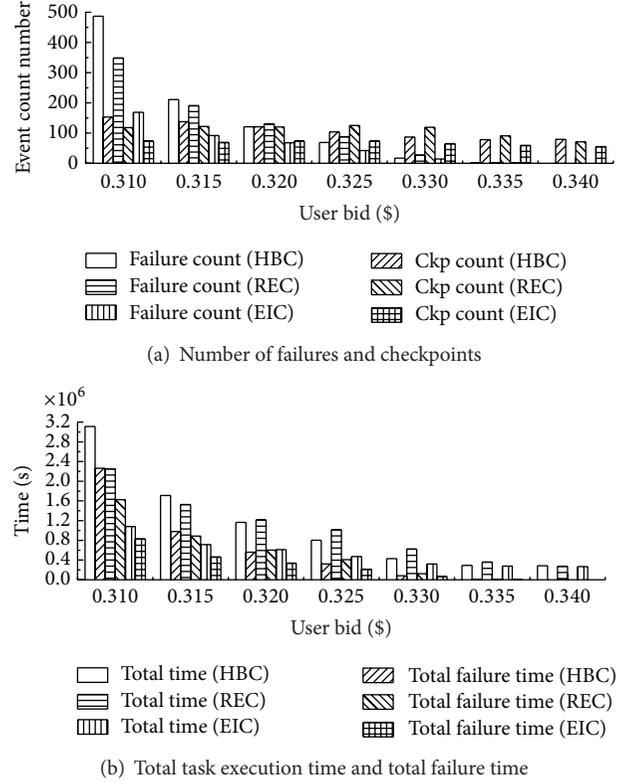


FIGURE 15: Performance comparison of checkpointing schemes according to user bid.

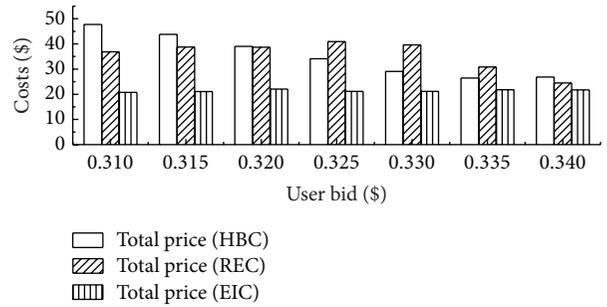


FIGURE 16: Comparison of total costs.

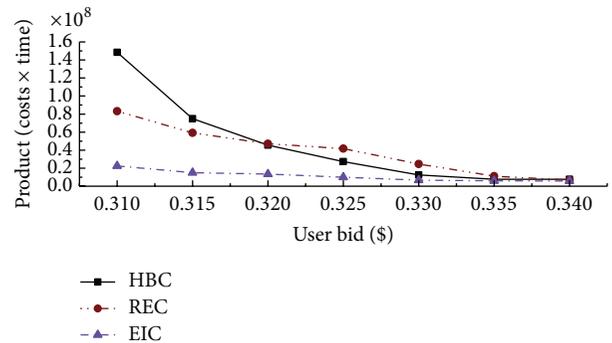


FIGURE 17: Comparison of combined metrics (total task execution time and costs).

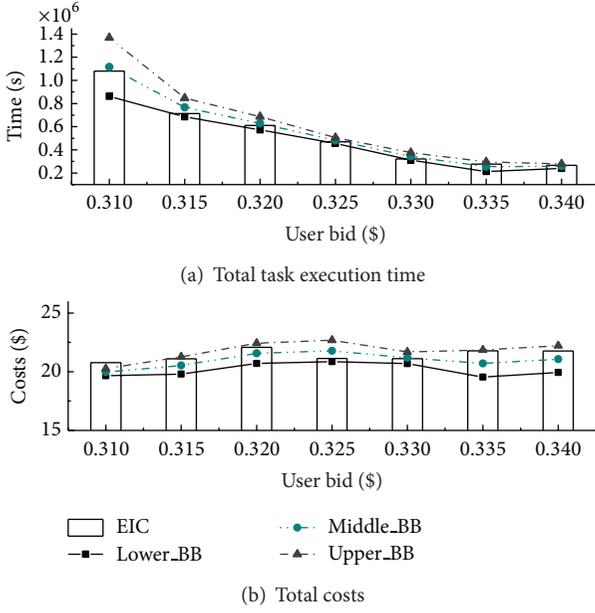


FIGURE 18: Comparison of actual EIC outputs (execution time and cost) and estimations according to the user bid.

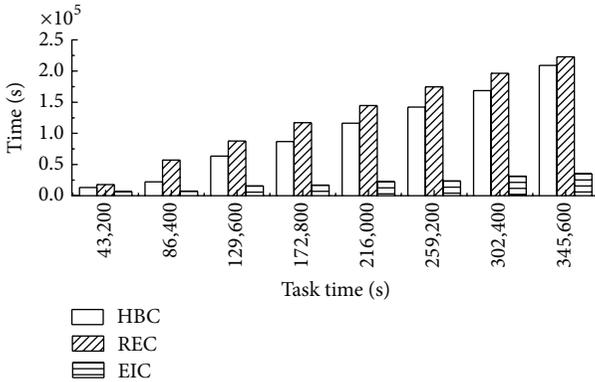


FIGURE 19: Comparison of rollback times according to the task time.

TABLE 3: Simulation parameters and values.

| Parameter          | Value       |
|--------------------|-------------|
| User bid           | \$0.32      |
| Checkpoint time    | 300 sec     |
| Recovery time      | 300 sec     |
| Minimum task time  | 43,200 sec  |
| Maximum task time  | 345,600 sec |
| Task time interval | 43,200 sec  |

lessens the rollback time on average by 80.61% and 84.36% over HBC and REC, respectively.

Figure 20 shows the performance comparison of EIC, HBC, and REC. Figures 20(a) and 20(b) show the numbers of failures and checkpoints, and total task execution time and total failure time according to the task time. The EIC reduces the number of checkpoints on average by 31.97% and 32.93%

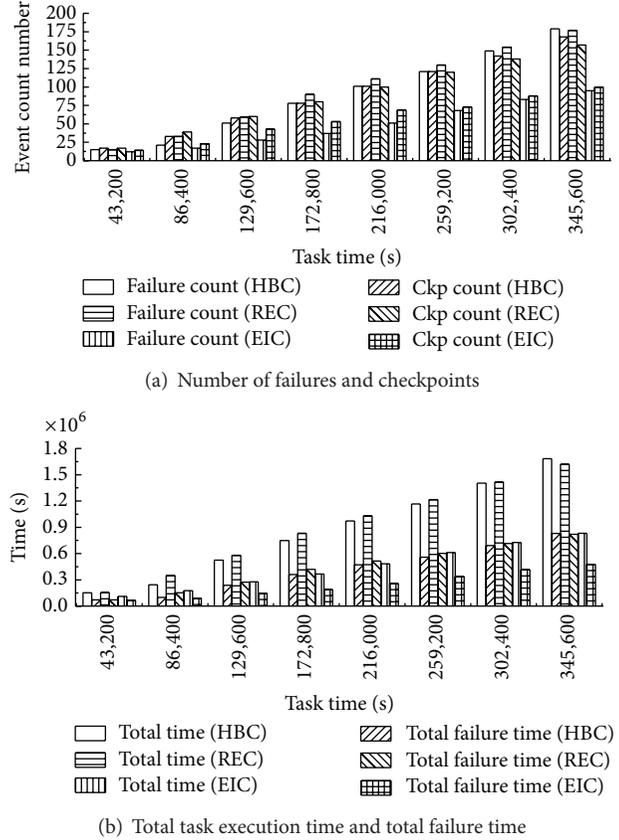


FIGURE 20: Performance comparison according to the task time.

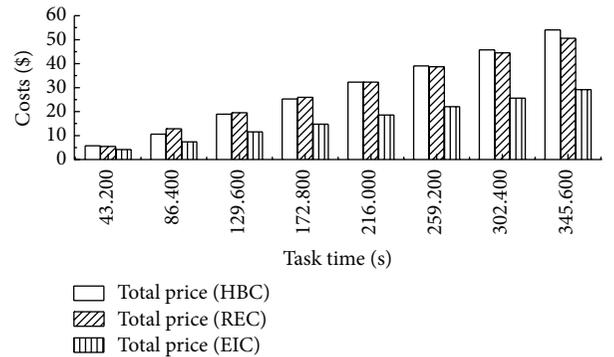


FIGURE 21: Comparison of total costs.

compared to HBC and REC, respectively. Thus, the EIC achieves performance improvements in the task execution time on average by 43.79% and 48.25% over HBC and REC, respectively.

Figure 21 shows the total cost according to the task time. The EIC reduces the cost on average by 39.38% and 40.08% compared to HBC and REC, respectively.

Figure 22 shows the combined performance metric, the product of the total task execution time, and cost. The rate of increase in the product in EIC is lowest among the compared schemes. The EIC achieves a performance improvement on

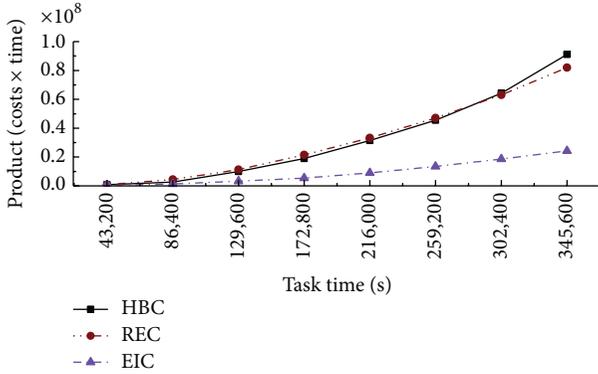


FIGURE 22: Comparison of combined metrics (product of total execution time and cost).

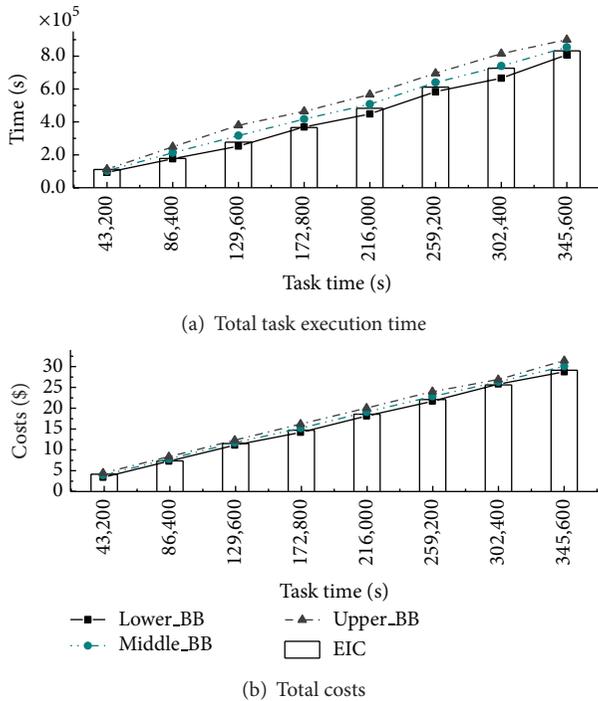


FIGURE 23: Comparison of actual EIC outputs (execution time and cost) and estimations according to the task time.

average by 65.36% and 68.51% when compared to HBC and REC, respectively.

Figure 23 shows the estimation accuracies according to the task time. The actual execution time and cost are located between the lower and upper bounds of the Bollinger Band. Figures 23(a) and 23(b) prove that the actual execution time and cost are close to the middle point of the Bollinger Band. They state that the EIC would be able to offer approximate ranges of total costs and task execution time to Cloud users.

Overall, the EIC significantly reduces the number of checkpoint trials compared to the existing checkpointing schemes. Furthermore, the rollback time is much lesser because the EIC adaptively performs the checkpointing operation according to the execution time and price. Simulation

results showed that our scheme achieved the cost-efficiency by reducing rollback time regardless of the resource types of spot instances.

Analyzing history to compute the estimated interval requires overheads such as CPU time. However, computations only involve failure probability, execution time and cost estimations, and a range of the Bollinger Band. Considering the advancement of modern computers, we strongly believe it would take the minimal amount of overheads for computations.

## 6. Conclusion

In this paper, we proposed the estimated interval-based checkpointing (EIC) in the unreliable cloud computing environment. The weighted moving average estimates the execution time and cost using the price history of spot instances to improve the performance and stability of task processing. The EIC performs the checkpointing operation, based on price and time thresholds. The thresholds are determined based on the moving average and the failure probability. They are used to determine the checkpointing position to recover from the potential failures of spot instances arising from the price fluctuation. The Bollinger Bands determines the lower and upper bounds of the estimated execution time and cost. The ranges are informed to users as guidance for their decision. The simulation results reveal that, compared to the hour-boundary checkpointing (HBC) and rising edge-driven checkpointing (REC), the EIC reduces the number of checkpoints by 35.97% and 37.92%, respectively, on average according to the user bid. It also reduces the rollback time by 72.46% and 88.49% on average. Consequently, the task execution time is decreased with ETC by 35.53% over HBC and 40.40% over REC. The EIC also provides the benefit of the cost reduction by 36.26% over HBC and 38.52% over REC, on average.

## 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) (NRF-2012RIA2A2A 02046684).

## References

- [1] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: vision, hype, and reality for delivering IT services as computing utilities," in *Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC '08)*, pp. 5–13, September 2008.
- [2] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Proceedings of the Grid Computing Environments Workshop (GCE '08)*, pp. 1–10, November 2008.

- [3] K. Mahajan, A. Makroo, and D. Dahiya, "Round Robin with server AF-finity: a VM load balancing algorithm for cloud based infrastructure," *Journal of Information Processing System*, vol. 9, no. 3, pp. 379–394, 2013.
- [4] M. M. Weng, T. K. Shih, and J. C. Hung, "A personal tutoring mechanism based on the cloud environment," *Journal of Convergence*, vol. 4, pp. 37–44, 2013.
- [5] A. Følstad, K. Hornbæk, and P. Ulleberg, "Social design feedback: evaluations with users in online ad-hoc groups," *Human-centric Computing and Information Sciences*, vol. 3, article 18, 2013.
- [6] H. N. Van, F. D. Tran, and J.-M. Menaud, "SLA-aware virtual resource management for cloud infrastructures," in *Proceedings of the 9th IEEE International Conference on Computer and Information Technology (CIT '09)*, pp. 357–362, October 2009.
- [7] Elastic Compute Cloud (EC2), 2014, <http://aws.amazon.com/ec2>.
- [8] GoGrid, 2014, <http://www.gogrid.com>.
- [9] FlexiScale, 2014, <http://www.flexiscale.com>.
- [10] I. Goiri, F. Julià, J. Guitart, and J. Torres, "Checkpoint-based fault-tolerant infrastructure for virtualized service providers," in *Proceedings of the 12th IEEE/IFIP Network Operations and Management Symposium (NOMS '10)*, pp. 455–462, April 2010.
- [11] S. Yi, D. Kondo, and A. Andrzejak, "Reducing costs of spot instances via checkpointing in the Amazon Elastic Compute Cloud," in *Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD '10)*, pp. 236–243, July 2010.
- [12] D. Jung, S. Chin, K. Chung, H. Yu, and J. Gil, "An efficient checkpointing scheme using price history of spot instances in cloud computing environment," in *Proceedings of the 8th IFIP International Conference on Network and Parallel Computing (NPC '11)*, pp. 185–200, 2011.
- [13] S. Yi, J. Heo, Y. Cho, and J. Hong, "Taking point decision mechanism for page-level incremental checkpointing based on cost analysis of process execution time," *Journal of Information Science and Engineering*, vol. 23, no. 5, pp. 1325–1337, 2007.
- [14] G. Singer, I. Livenson, M. Dumas, S. N. Srirama, and U. Norbistrath, "Towards a model for cloud computing cost estimation with reserved resources," in *Proceedings of the 2nd ICST International Conference on Cloud Computing (CloudComp '10)*, Springer, Barcelona, Spain, October 2010.
- [15] M. Mazzucco and M. Dumas, "Reserved or on-demand instances? A revenue maximization model for cloud providers," in *Proceedings of the 4th IEEE International Conference on Cloud Computing (CLOUD '11)*, pp. 428–435, July 2011.
- [16] W. Voorsluys and R. Buyya, "Reliable provisioning of spot instances for compute-intensive applications," in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications (AINA '12)*, pp. 542–549, March 2012.
- [17] Q. Zhang, E. Gürses, R. Boutaba, and J. Xiao, "Dynamic resource allocation for spot markets in clouds," in *Proceedings of the 11th USENIX Conference on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE '11)*, pp. 1–6, 2011.
- [18] Amazon EC2 spot Instances, 2014, <http://aws.amazon.com/ec2/spot-instances>.
- [19] Cloud Exchange, 2014, <http://cloudexchange.org>.
- [20] A. Andrzejak, D. Kondo, and S. Yi, "Decision model for cloud computing under SLA constraints," in *Proceedings of the 18th Annual IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '10)*, pp. 257–266, August 2010.
- [21] P. Patel, A. Ranabahu, and A. Sheth, "Service level agreement in cloud computing," in *Proceedings of the Conference on Object Oriented Programming Systems Languages and Applications*, pp. 212–217, 2009.
- [22] G. Dagnino, "Technical analysis, the markets and moving averages," Tech. Rep., The Peter Dag Portfolio Strategy & Management, 2013.
- [23] R. J. Hyndman, "Moving averages," Tech. Rep., Department of Econometrics and Business Statistics, Monash University, 2009.
- [24] J. Bollinger, *Bollinger on Bollinger Bands*, McGraw Hill, 2002.
- [25] Daytrader, "Bollinger bands as an entry technique," 2000.

## Research Article

# Whitelists Based Multiple Filtering Techniques in SCADA Sensor Networks

**DongHo Kang,<sup>1</sup> ByoungKoo Kim,<sup>1</sup> JungChan Na,<sup>1</sup> and KyoungSon Jhang<sup>2</sup>**

<sup>1</sup> *Convergence Security Research Section, Electronics and Telecommunications Research Institute (ETRI), Daejeon 305-700, Republic of Korea*

<sup>2</sup> *Department of Computer Engineering, Chungnam National University, Daejeon 305-764, Republic of Korea*

Correspondence should be addressed to DongHo Kang; [dhkang@etri.re.kr](mailto:dhkang@etri.re.kr)

Received 31 January 2014; Accepted 6 May 2014; Published 28 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 DongHo 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.

Internet of Things (IoT) consists of several tiny devices connected together to form a collaborative computing environment. Recently IoT technologies begin to merge with supervisory control and data acquisition (SCADA) sensor networks to more efficiently gather and analyze real-time data from sensors in industrial environments. But SCADA sensor networks are becoming more and more vulnerable to cyber-attacks due to increased connectivity. To safely adopt IoT technologies in the SCADA environments, it is important to improve the security of SCADA sensor networks. In this paper we propose a multiple filtering technique based on whitelists to detect illegitimate packets. Our proposed system detects the traffic of network and application protocol attacks with a set of whitelists collected from normal traffic.

## 1. Introduction

In general, a SCADA network is a network required for effective remote monitoring and control of the devices remotely scattered. These networks interlink and operate the SCADA systems and various controllers needed to monitor field devices in real-time. In the past, SCADA networks operated in close environments isolated from external networks and adopted an undisclosed protocol and software in order to monitor and control various field devices internally. But modern SCADA systems have distributed architecture and are connected to the corporate network and to the Internet. Recently IoT technologies begin to merge with SCADA sensor networks to more efficiently gather and analyze real-time data from sensors in industrial environments. In addition, these systems use general-purpose operation systems and industry-standard communication protocols such as Modbus and DNP3 for communication between a SCADA system and field devices such as programmable logic controller (PLC) and remote terminal unit (RTU). The increased connectivity and the use of standard protocols can help to optimize manufacturing and distribution processes. But, they also expose these networks to the myriad security problems

of the Internet [1]. Before we describe our approach we first introduce the SCADA architecture and protocol for understanding SCADA systems.

*1.1. The SCADA Architecture.* SCADA networks come in various forms and layers according to the target and size. SCADA networks are employed in many industrial domains including manufacturing and electricity generation. In the past, they were isolated from other networks and proprietary protocols and software were adopted to monitor and control the various local devices [2]. Hence, security services in these networks were considered to be unlikely. But, due to the adoption of Ethernet and TCP/IP, they have evolved an architecture strongly based on connectivity to improve efficiency and productivity. The SCADA architecture usually consists of three different domains [3]. A typical SCADA architecture is shown in Figure 1.

A control center includes human machine interface (HMI), SCADA servers, and historian systems for process control, the gathering of data in real-time from field devices in order to control sensors and actuators. A field site includes multiple field devices that send commands to actuators and

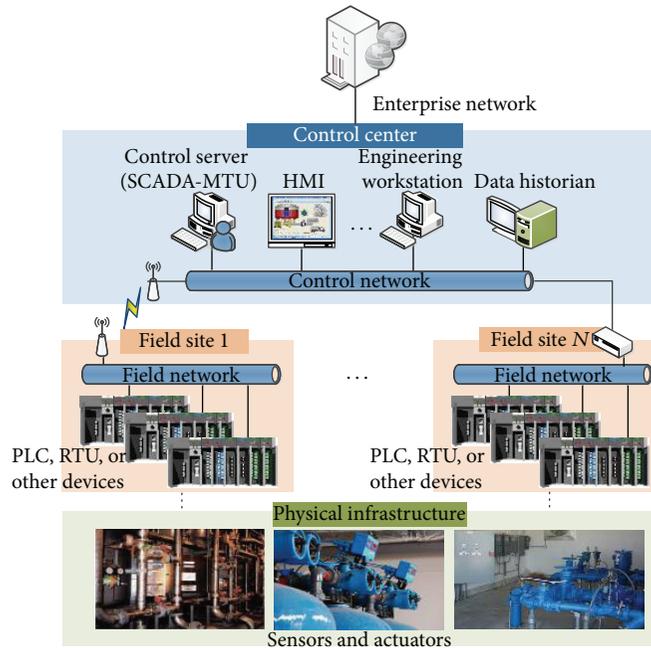


FIGURE 1: SCADA system general layout.

provide the data received from sensors to SCADA servers. Physical infrastructure consists of many different types of sensors and actuators that are monitored and controlled by a field device.

*1.2. Modbus Protocol.* Modbus is an application layer messaging protocol which provides master/slave communication between devices in SCADA systems [4]. The function code of Modbus informs the slave of what type of action to perform. For instance, Modbus function “0 × 01” can be used to read the status of an output in the Modbus slave device.

Figure 2 shows the communication between devices connected on the Modbus TCP/IP network. The master that initiates a Modbus transaction builds the Modbus application data unit (ADU). The Modbus ADU consists of the Modbus application protocol (MBAP) header and the protocol data unit (PDU). The PDU has a function code and function parameters. The function codes indicate to the slave which kind of action to perform. The Modbus TCP/IP uses the TCP/IP stack for communication and extends the PDU with an IP header. But there are no security functions in the protocol. The simplicity of the Modbus protocol makes it relatively simple to attack Modbus slaves [5]. If any attackers have broken into the Modbus master, they may send illegal commands to Modbus slaves to perform abnormal behaviors.

The purpose of this paper is to discuss our approach and confirm the validity of our proposed system for preventing network and application protocol attacks in SCADA sensor networks. This paper is organized as follows. Section 2 gives detailed cyber threats. Section 3 describes a detailed explanation of our proposed system. Section 4 presents related works and Section 5 gives conclusion.

TABLE 1: Network protocol attacks.

| Attack type                       | Attacks                |
|-----------------------------------|------------------------|
| Host discovery                    | OS fingerprinting      |
| Scan                              | TCP SYN/ACK scan       |
|                                   | TCP connect() scan     |
|                                   | TCP FIN stealth scan   |
|                                   | Xmas tree stealth scan |
|                                   | TCP null stealth scan  |
|                                   | Windows scan           |
| DoS attack<br>(Denial-of-service) | RPC scan               |
|                                   | Version detection scan |
|                                   | TCP/UDP flooding       |
|                                   | Smurf attack           |

## 2. Cyber Threats in SCADA Networks

We surveyed vulnerability assessment tools, Metasploit [6], Nessus [7], and Modscan [8] for the classification of cyber-attacks in SCADA networks. These tools are commonly available to find known and newly discovered vulnerabilities on SCADA systems. And we surveyed some reports that were released by the projects of DigitalBond [9, 10]. As a result of our survey, we describe that various types of attacks on SCADA systems can be grouped into two categories: network protocol attacks and application protocol attacks.

*2.1. Network Protocol Attacks.* Most network protocol based attacks happened in Internet environment may be caused in SCADA networks were adopted IP network. These types of attacks use weak points of network protocols such as TCP/IP suite that have a number of serious security flaws. We introduce some types of network protocol attacks. Table 1

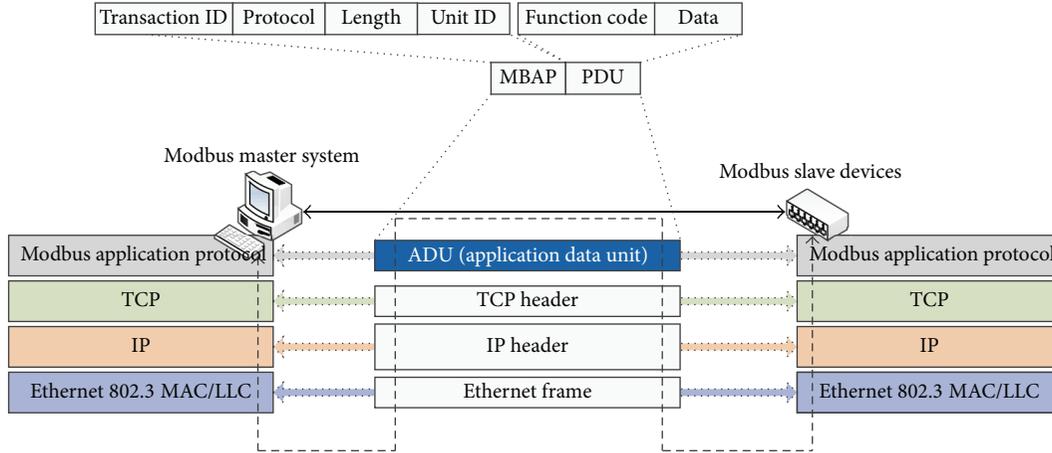


FIGURE 2: The format of Modbus TCP/IP ADU.

TABLE 2: Application protocol attacks.

| Attack type                | Attacks                                |
|----------------------------|----------------------------------------|
| Application scan           | Modbus version scanner                 |
|                            | PLC Modbus mode identification         |
|                            | PLC IO scan status                     |
|                            | Report slave ID                        |
|                            | Function code scan                     |
| Improper command execution | Force listen only mode                 |
|                            | Read/write request to a PLC            |
|                            | Slave device busy exception code delay |
|                            | Acknowledge exception code delay       |
|                            | Broadcast request from a client        |

shows types of network protocol attacks. Host discovery is the process for gathering information about each host such as its operating system and version to verify whether they can be accessed or not. Using the information gathered about each target host in the host discovery step, attackers launch scan to conform what ports are open, with listening services on target systems. Host discovery and scan attack are the common type of passive attacks to collect the fundamental information of vulnerabilities on target systems. Denial-of-service (DoS) attack is active attack to make systems or network resource unavailable. Network protocol attacks have two characteristics as follows.

- (i) Random access: host discovery or scan attacks generally send packets with the sequential or random destination addresses and ports to target networks or systems for obtaining the list of target systems and their services.
- (ii) Source address spoofing: DoS attack does not consider receiving responses to the attack packets. Therefore, attackers can send packets with a forged source IP address for obscuring the true source of the attack.

2.2. *Application Protocol Attacks.* In our work, we surveyed Modbus/TCP as an application protocol. Application protocol attacks can cause damage to field devices, being controlled

by sending out improper commands, because they do not support integrity checking and authentication mechanism. Like network protocol attacks, these attacks also preceded by a step of gathering information about devices for finding vulnerable targets in a network. Table 2 shows generally types of application protocol attacks.

Application protocol attacks have the following characteristic.

*Unpredictable Command.* SCADA systems generally produce predictable sets of command used for communication between a SCADA server and field devices. On the contrary, application protocol attacks tend to use unconventional commands at irregular interval.

### 3. Our Proposed System: The IndusCAP-Gate System

Our proposed system, the so-called IndusCAP-Gate system, automatically generates whitelists by analyzing the traffic and performs multiple filtering based on whitelists for blocking against unauthorized access from external networks. Figure 3 shows the packet processing flows of the IndusCAP-Gate system.

In the analysis phase the system performs the process of packet decoding and extracts data parameters in the captured traffic for building whitelists. After the analysis phase has been completed, the multiple filters inspect all incoming packets to detect abnormal behavior based on whitelists in the detection phase.

3.1. *The Analysis Phase.* The analysis phase is an initial training stage for building whitelists. The IndusCAP-Gate system captures and analyzes the traffic on communication between SCADA servers and field devices. The phase is executed for a predefined period and generates whitelists by analyzing normal SCADA traffic. Whitelists are the set of policies to help determine whether incoming packets from

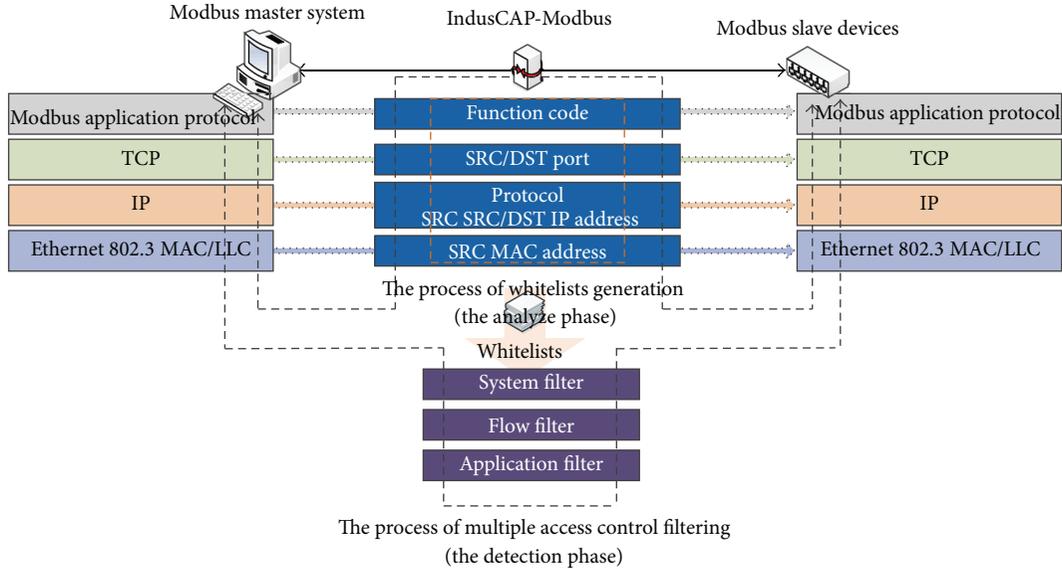


FIGURE 3: The packet processing flows of the IndusCAP-Gate system.

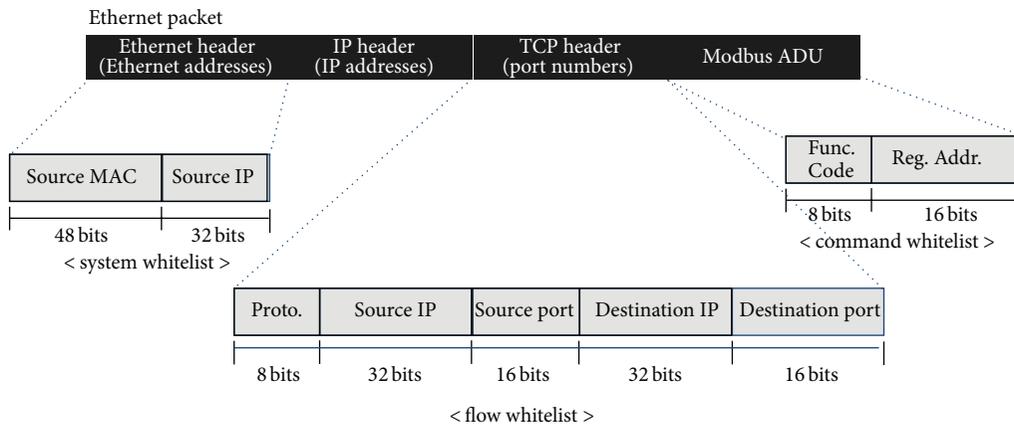


FIGURE 4: The format of whitelists.

external networks are abnormal. They have three types and Figure 4 shows the format of whitelists.

The system whitelist has multiple source MAC/IP address pairs. We treat these pairs as authenticated systems during the detection phase. The flow whitelist contains a set of the 5-tuple (i.e., the source and destination IP address, the same source and destination port, and the same protocol) information. The whitelists are referred to the flow filter in order to identify the abnormal flows. The command whitelist is used to detect unauthorized Modbus commands by the command filter. Upon completion of the phase, the detection phase uses the result of the analysis phase to identify abnormal traffic. Each whitelist maintained by the system that monitors incoming packets will add entries without a need for human intervention. We assume that the traffic gathered in the analysis phase includes only normal data that does not contain packets generated by the attack. Since SCADA networks, unlike conventional networks, have relatively limited connections to outside networks, attack

attempts do not occur frequently and the analysis phase is executed only for a defined short period after the initial installation. We are confident that the assumption will be valid for our approach.

3.2. *The Detection Phase.* The IndusCAP-Gate system provides whitelists based multiple filters to block unauthorized access to field devices in field networks. The system is positioned between SCADA network and field networks. Figure 5 shows the system architecture.

The IndusCAP-Gate system was designed to protect field devices from various cyber-attacks. For archiving the purpose, the system consists of four functions. The packet collection and control function perform the role of forwarding or blocking packets according to the result of multiple filters. The network layer access control function determines whether to drop or route the packet by inspecting the Ethernet, IP, TCP, and UDP headers. If the incoming packet

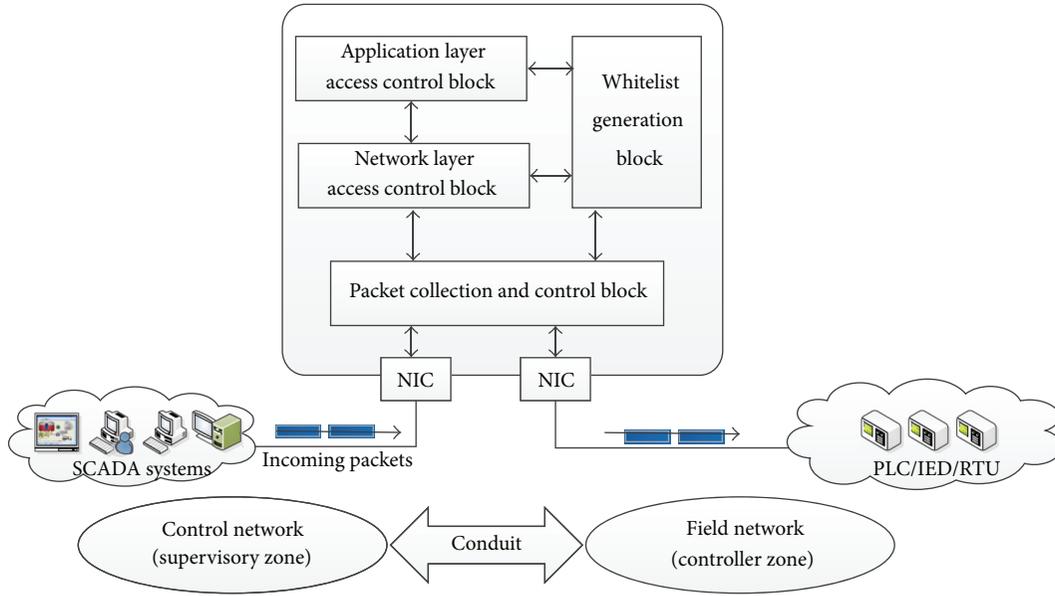


FIGURE 5: The system architecture.

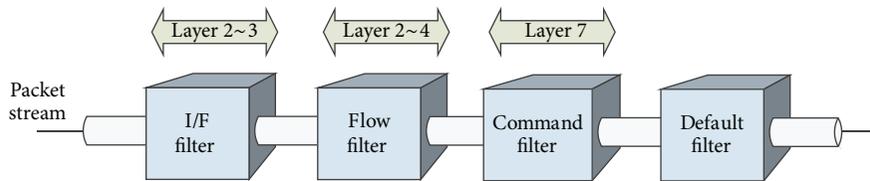


FIGURE 6: The process of multiple filters.

can meet a condition in system whitelist or the flow whitelist, the function routes it using the system filter or the flow filter. The application layer access control function performs application-level access control at the application layer. This function analyzes the incoming packets using the command filter and then blocks unauthorized access to the command. Using the functions above, the IndusCAP-Gate system blocks unauthorized access from illegitimate traffic.

3.3. Multiple Access Control Filter Based Blocking of Unauthorized Access. As described above, the IndusCAP-Gate system's multiple filters consist of 4 filters.

Figure 6 shows the process of multiple filters. Each filter can be described as follows.

- (i) Default filter: a default filter is enabled according to the existence of policy of other access control filters (disabled if there is at least one policy of other access control filters for each interface). It only decides whether the incoming packet will be allowed or denied. Since such enables total access control of incoming packets into a specific interface, it can be useful for special-purpose access control.
- (ii) System filter(I/Ffilter): the system whitelist, the policies of MAC/IP pair, is applied for each interface.

Only those packets conforming to the applied policies are selected and delivered to the opposite interface.

- (iii) Flow filter: the filter performs 5-tuple-based access control with the flow whitelist at the network layer.
- (iv) Command filter: the filter performs application-level access control and analyzes the Modbus protocol. It controls access to the command with the command whitelist.

Figure 7 shows the overall packet processing flows of multiple access control filters. As shown in the figure, processing of incoming packets into the interfaces is the same except for those branching into each interface. Only the packets allowed through a filter can be delivered to the next filter. In other words, only those packets allowed through all filters are delivered to the opposite interface. The process allows the IndusCAP-Gate system to block unauthorized access to the control system and apply access control policies efficiently according to the size and nature of the control system intranet.

The IndusCAP-Gate system was implemented to run in Linux OS, adopting the UNO-3072L platform to suit the nature of the SCADA environment. The packet processing performance of the IndusCAP-Gate system was tested using the IXIA traffic generator. Since the SCADA networks generally have low bandwidth, up to 20 Mbps packets transfers



phase. The system can assist security administrators in identifying normal traffic by generating whitelists and decrease false positives. After the analysis phase has been completed, the proposed system inspects the traffic on communication between SCADA systems and field devices with whitelists. Our proposed system may effectively prevent unknown attacks using whitelists.

In future work, we plan to extend this work with network behavior based anomaly detection technique for detecting anomalous SCADA traffic. And then we intend to apply the approach in the other networks.

## Conflict of Interests

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

## Acknowledgment

This paper is extended and improved form accepted paper of KCIC-2013/FCC-2014 conferences. And this work was supported by the IT R&D program of MSIP/KEIT [010041560, a development of anomaly detection and multilayered response technology to protect an intranet of a control system for the availability of pipeline facilities].

## References

- [1] K. Stouffer, J. Falco, and K. Scarfone, *Guide to Industrial Control Systems (ICS) Security*, NIST Special Publication 800. 82, 2008.
- [2] B. Galloway and G. P. Hancke, "Introduction to industrial control networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 2, pp. 860–880, 2013.
- [3] V. M. Iquire, S. A. Laughter, and R. D. Williams, "Security issues in SCADA networks," *Computers & Security*, vol. 25, no. 7, pp. 498–506, 2006.
- [4] I. D. A. Modbus, "Modbus application protocol specification v1. 1a," North Grafton Grafton, Mass, USA, 2004, <http://www.modbus.org/specs.php>.
- [5] <http://www.digitalbond.com/scadapedia/protocols/modbus-2/>.
- [6] <http://www.metasploit.com/>.
- [7] <http://www.tenable.com/products/nessus>.
- [8] <https://code.google.com/p/modscan/>.
- [9] <http://www.digitalbond.com/tools/basecamp/>.
- [10] <http://www.digitalbond.com/tools/quickdraw/>.
- [11] H.-I. Kim, Y.-K. Kim, Y.-K. Kim, and J.-W. Chang, "A grid-based cloaking area creation scheme for continuous LBS queries in distributed systems," *Journal of Convergence*, vol. 4, no. 1, pp. 23–30, 2013.
- [12] M. Yoon, Y.-K. Kim, and J.-W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, pp. 15–22, 2013.
- [13] A. Sinha and D. Krishan Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013.
- [14] M. I. Malkawi, "The art of software systems development: reliability, avail-ability, maintainability, performance (RAMP)," *Human-Centric Computing and Information Sciences*, vol. 3, article 22, 2013.
- [15] J. W. K. Gnanaraj, K. Ezra, and E. B. Rajsingh, "Smart card based time efficient authentication scheme for global grid computing," *Human-Centric Computing and Information Sciences*, vol. 3, article 16, 2013.
- [16] H.-R. Lee, K.-Y. Chung, and K. -S. Jhang, "A study of wireless sensor network routing protocols for maintenance access hatch condition surveillance," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 237–246, 2013.
- [17] K. Peng, "A secure network for mobile wireless service," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 247–258, 2013.
- [18] D.-K. Kwon, K. Chung, and K. Choi, "A dynamic zigbee protocol for reducing power consumption," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 41–52, 2013.
- [19] M. M. Weng, T. K. Shih, and J. C. Hung, "A personal tutoring mechanism based on the cloud environment," *Journal of Convergence*, vol. 4, pp. 37–44, 2013.
- [20] P. Oman and M. Phillips, "Intrusion detection and event monitoring in SCADA networks," in *Critical Infrastructure Protection*, pp. 161–173, Springer, New York, NY, USA, 2007.
- [21] T. H. Morris, B. A. Jones, R. B. Vaughn, and Y. S. Dandass, "Deterministic intrusion detection rules for MODBUS protocols," in *Proceedings of the 46th Annual Hawaii International Conference on System Sciences (HICSS '13)*, pp. 1773–1781, Wailea, Hawaii, USA, January 2013.
- [22] <http://www.snort.org>.
- [23] S. Cheung, B. Dutertre, M. Fong et al., "Using model-based intrusion detection for SCADA networks," in *Proceedings of the SCADA Security Scientific Symposium*, 2007.
- [24] P. Düssel, C. Gehl, P. Laskov et al., "Cyber-critical infrastructure protection using real-time payload-based anomaly detection," in *Critical Information Infrastructures Security*, pp. 85–97, Springer, Berlin, Germany, 2010.
- [25] R. R. R. Barbosa, R. Sadre, and A. Pras, "A first look into SCADA network traffic," in *Proceedings of the IEEE Network Operations and Management Symposium (NOMS '12)*, pp. 518–521, Maui, Hawaii, USA, April 2012.
- [26] V. A. Siris and F. Papagalou, "Application of anomaly detection algorithms for detecting SYN flooding attacks," *Computer Communications*, vol. 29, no. 9, pp. 1433–1442, 2006.

## Research Article

# TSMC: A Novel Approach for Live Virtual Machine Migration

**Jiaxing Song, Weidong Liu, Feiran Yin, and Chao Gao**

*Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China*

Correspondence should be addressed to Jiaxing Song; [jxsong@tsinghua.edu.cn](mailto:jxsong@tsinghua.edu.cn)

Received 23 January 2014; Accepted 6 May 2014; Published 20 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Jiaxing Song 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.

Cloud computing attracted more and more attention in recent years, and virtualization technology is the key point for deploying infrastructure services in cloud environment. It allows application isolation and facilitates server consolidation, load balancing, fault management, and power saving. Live virtual machine migration can effectively relocate virtual resources and it has become an important management method in clusters and data centers. Existing precopy live migration approach has to iteratively copy redundant memory pages; another postcopy live migration approach would lead to a lot of page faults and application degradation. In this paper, we present a novel approach called TSMC (three-stage memory copy) for live virtual machine migration. In TSMC, memory pages only need to be transmitted twice at most and page fault just occurred in small part of dirty pages. We implement it in Xen and compare it with Xen's original precopy approach. The experimental results under various memory workloads show that TSMC approach can significantly reduce the cumulative migration time and total pages transferred and achieve better network IO performance in the same time.

## 1. Introduction

After the wave of pervasive computing and grid computing [1–3], the conception of cloud computing was officially proposed by Google. Since it appeared, cloud computing has a huge impact on the entire IT industry. There are many hot research areas in cloud computing. For example, resource management [4, 5] becomes more important in cloud computing. A lot of research works have been done worldwide [6–9].

Virtualization technology has played a very vital role in resource management of cloud computing and it develops rapidly in recent years. The resources of a single physical machine are divided into multiple isolated virtual resources by using some virtualization softwares [10]. The isolated virtual environment is called virtual machine (VM) [11]. It can provide application isolation, server consolidation, better multiplexing of data center resources, the ability to flexibly remap physical resources, and so on [12].

Live migration is the key point of virtualization technologies. It allows VMs fast relocation in data center and nonawareness of downtime. Lots of live migration techniques have been brought up these years [13, 14]. Most of them use precopy approach. It first transfers all memory pages

to the target VM and then copies pages which are dirtied iteratively. However, great service degradation would happen in precopy phase because migration daemon continually consumes network bandwidth to transfer dirty pages in each round. Another approach called postcopy is also introduced into live migration of VMs. In this approach, all memory pages are transferred only once during the whole migration process and the baseline total migration time is achieved. But the downtime is much higher than that of precopy due to the latency of fetching pages from the source node before VM can be resumed on the target.

In this paper, we present an optimized memory copy approach for live virtual machine migration. We combine the advantages of active pushing and on-demand copy; first copy all memory pages to target and record dirty bitmap in this phase (full memory copy stage), then suspend the VM, transmit CPU state and dirty bitmap (dirty bitmap copy stage), and finally resume the new VM and copy dirty pages from source to target (dirty page copy stage). We call it TSMC (three-stage memory copy). The main goal of TSMC is to minimize total migration time and reduce network traffic. Most of the memory pages need to be copied once in full memory copy stage; only dirtied pages need to be copied twice. Many approaches have been proposed to

evaluate the performance of virtualization [15]. We chose to implement this TSMC approach on Xen [16] and compared it with original precopy method in Xen. The experiment results under various memory workloads show that our approach can significantly reduce the cumulative migration time and total pages transferred.

This paper is organized as follows. In Section 2, we describe related work. Then, in Section 3, we describe the design and implementation of TSMC and we present the experimental results in Section 4. Finally, we make a conclusion in Section 5.

## 2. Related Work

Precopy [17] live virtual machine migration approach was firstly proposed. In precopy approach, it first transfers all memory pages and then copies pages just modified during the last round iteratively, until writable working set (WWS) becomes small or the preset number of iterations is reached. Eventually, it suspends VM in source node and sends CPU state and the remaining dirty pages in the last round to the target, where the VM is restarted. There are many virtualization platforms using this approach, such as Xen [16], KVM [18], and VMware [19]. Precopy is the prevailing live migration technique to perform live migration of VMs, but in write-intensive workloads, memory pages will be repeatedly dirtied and may have to be transmitted multiple times.

Postcopy [20] instead of precopy was proposed to solve this problem and reduce total migration time. Postcopy migration defers the memory transfer phase until after the VM's CPU state has already been transferred to the target and resumed there. Postcopy ensures that each memory page is transferred at most once, thus avoiding the duplicate transmission overhead of precopy. But the downtime is much higher than that of the precopy due to the latency of fetching pages from the source node before VM can be resumed on the target.

Jin et al. [21] proposed a new mechanism using adaptive compression of migrated data; different compression algorithms are chosen depending on characteristics of memory pages. They first used memory compression to provide fast VM migration and they also designed a zero-aware characteristics-based compression (CBC) algorithm for live migration. In the source node, data being transferred in each round are first compressed by their algorithm. When arriving on the target, compressed data are then decompressed. However, memory compression increases the system overhead.

To overcome the shortcomings of precopy and postcopy approaches, many other live migration methods [22, 23] are proposed, but almost all of them have their own limitations.

## 3. Design and Implementation

In this section, we introduce the phase of live migration and describe the design of TSMC approach and its implementation on Xen. The performance of any live virtual machine migration strategy could be gauged by the following metrics.

*Downtime.* The time during which the migrating VMs are not executed.

*Readiness Time.* The time between the start of migration and the start of downtime.

*Recover Time.* The time between resuming the VMs execution at the target and the end of migration.

*Total Migration Time.* The total time of all migration times from start to finish.

*Pages Transferred.* The total amount of memory pages transferred, including duplicates, across all periods.

*3.1. Memory Migration Phases.* Efficient synchronization of the memory state is the key issue of live virtual machine migration. Memory transfer can be achieved by following three phases [17].

*Push.* The source VM continues running while certain pages are pushed across the network to the new destination. To ensure consistency, pages modified during this process must be resent.

*Stop-and-Copy.* The source VM is stopped, pages are copied across to the destination VM, and then the new VM is started.

*Pull.* The new VM is executed and, if it accesses a page that has not yet been copied, this page is faulted in "pulled" across the network from the source VM.

Figure 1(a) shows precopy approach; it combines push copying and stop-and-copy. Another approach called postcopy in Figure 1(b) uses stop-and-copy and pull copying.

*3.2. Design of TSMC.* To solve the weaknesses of existing live migration methods, we propose a new approach called three-stage memory copy (TSMC) which combines three phases of memory transfer. The entire memory synchronization is divided into three stages. Figure 1(c) is the three-stage copy timeline.

*Full Memory Copy.* Copy all memory pages from source VM to destination when the source VM continues running and record pages modified during this process.

*Dirty Bitmap Copy.* Suspend source VM, copy recorded dirty bitmap to target node, and mark corresponding pages as dirty in destination VM.

*Dirty Pages Copy.* Resume new VM and then active push or copy dirty pages on demand from source VM to destination.

Compared with precopy, three-stage copy avoids iterative copy of dirty pages; most of the memory pages are just copied once and only dirtied pages in full memory copy stage need to be copied twice. It significantly reduces pages transferred, thus reducing the usage of network bandwidth. Meanwhile, only dirty bitmap and CPU state need to be transferred in suspend phase; downtime of VM is also shortened. Although

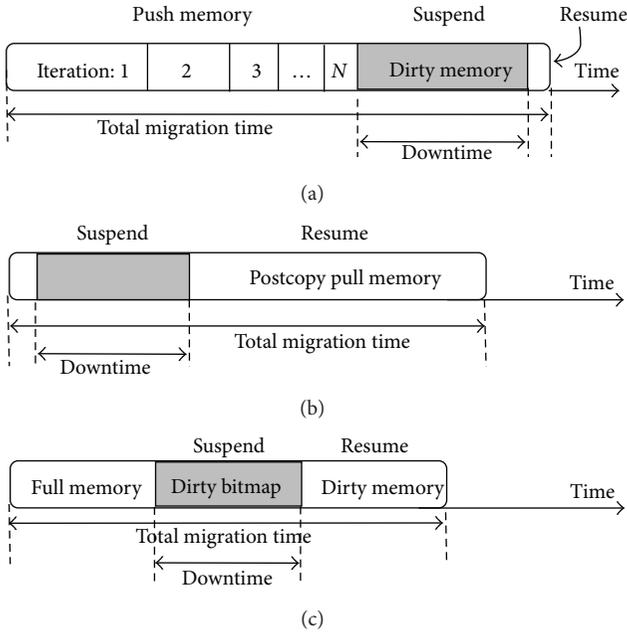


FIGURE 1: Timeline for live migration approach.

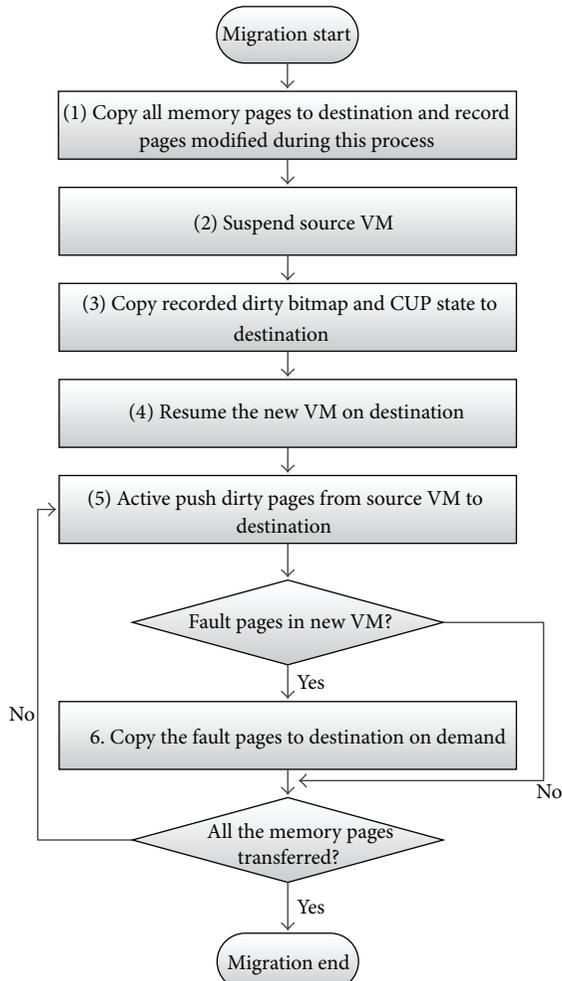


FIGURE 2: Procedure of TSMC.

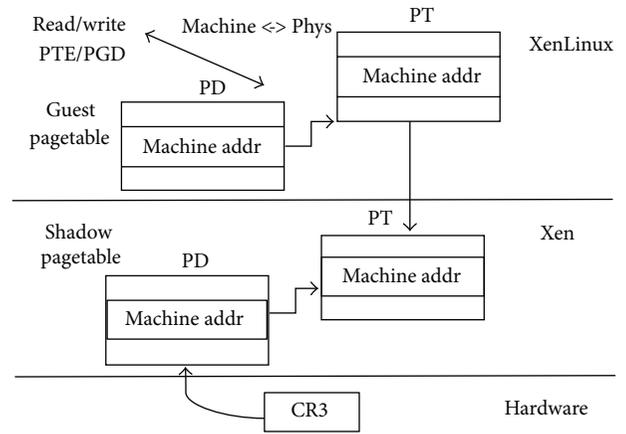


FIGURE 3: Shadow page table.

it would be interrupted in dirty pages copy stage because of page fault, but relative to full memory copy after resuming new VM in postcopy approach, three-stage copy just transfers dirtied pages after resuming, which significantly reduces the page fault rate and avoids obvious application degradation; also, it shortens the duration of the migration.

There are two methods used for transferring dirty page: on-demand copy and active push. Once the VM is resumed on the target, page faults would happen when memory access dirtied page; it can be serviced by requesting the referenced page over the network from the source node. However, page faults in new VM are unpredictable; on-demand copy would lead to longer resume time, so we combine it with active push whose source host periodically pushes dirty pages to the target in a preset time interval.

The procedure of TSMC is shown in Figure 2. In full memory copy phase, the update of memory pages should be recorded to dirty pages bitmap. Otherwise, memory changes of applications during the process cannot be updated to the new VM. Operating system (OS) on VM maintains the mapping page table from VM's linear addresses to OS's physical addresses, while VM monitor maintains translation page table from VM's physical addresses to physical host's physical addresses. VM monitor cannot monitor the memory changes of VM directly due to the transparency demand. So we utilize translation page table in virtualization tools to monitor the update of pages. In dirty pages copy phase, the page faults also need to be captured by VM monitor, which also can be achieved in the same way.

On-demand copy is the easiest and the slowest way. When the VM on destination resumes, the page faults will be transferred to source VM via the network and request the corresponding memory pages. Although on-demand copy copies dirty pages only once, it lengthens recovery time and degrades software performance. So it is unacceptable to transfer memory pages using on-demand copy alone.

Active push can reduce recovery time efficiently. It also reduces the long-time occupation of source VM's resources. After new VM resumes, active push pushes dirty pages from source VM to destination in a preset interval. It avoids some page faults on new VM. When page faults occur, we request

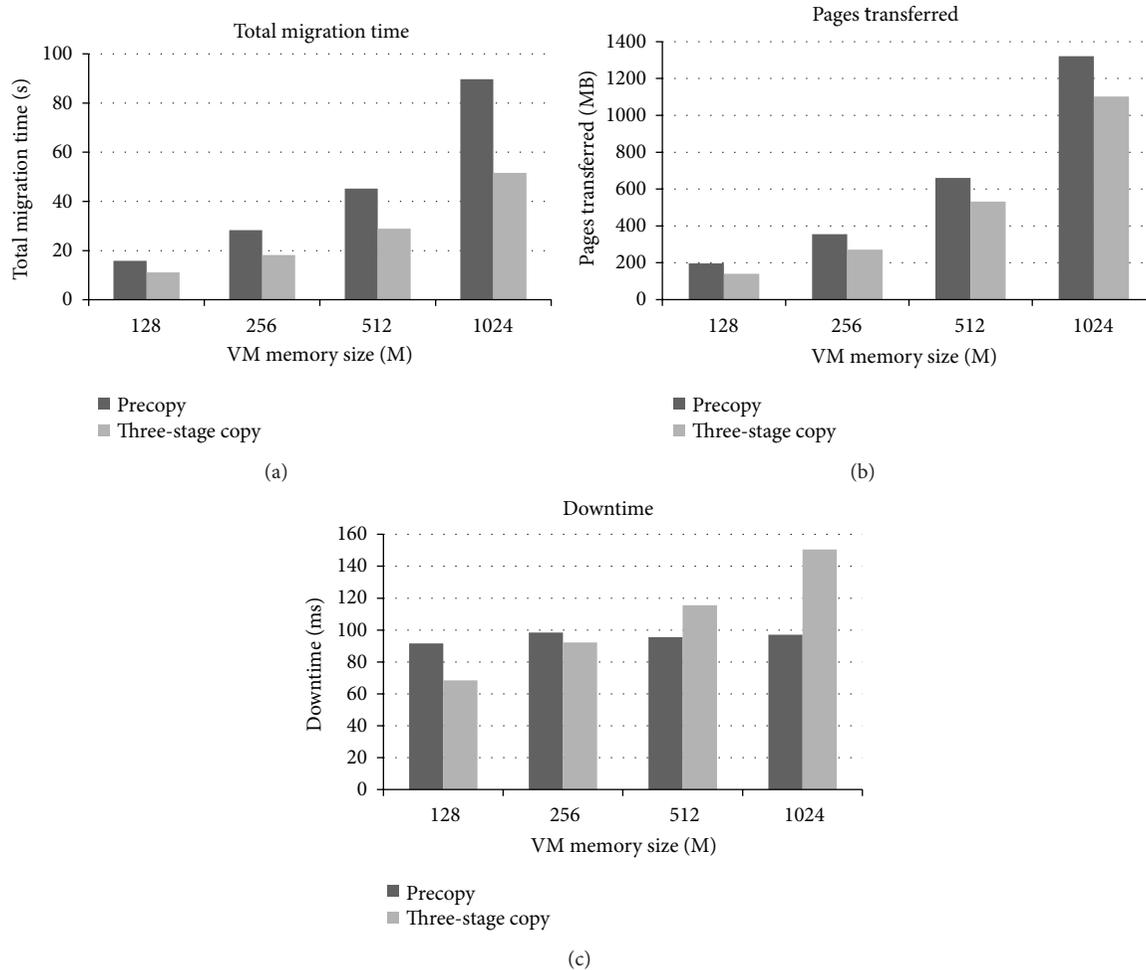


FIGURE 4: Comparison of total migration time, pages transferred, and downtime.

pages from source VM in the way of on-demand copy. The performance will be improved greatly by combining on-demand copy and active push.

Prepull was first brought up to predict the recent working set of softwares and it was based on software's running history. In three-stage copy, prepull is used to predict page faults in new VM. When page faults occur, then the pages around the missing page will probably be accessed, which leads to another page fault. Prepull increases the memory transfer window. When requesting pages from source VM, it transfers the pages around along with the request pages. In this way, less page faults will occur in the future.

**3.3. Implementation on Xen.** We implemented TSMC on Xen 4.1.4. The point of our approach is to capture and recode dirty pages. Shadow page tables are used by Xen's hypervisor to keep track of the memory state of guest OS; it can be used to capture dirty pages. Figure 3 shows the process of shadow page table. Shadow page tables are a set of read-only page tables for each VM maintained by the hypervisor that maps the VM's memory pages to the physical frames. Actually, it is equivalent to a backup of the original page tables; any

updates in guest OS's page table will notify Xen's hypervisor by Hypercall.

Because all page tables in guest OS are mapped to read-only shadow page tables, any updates in page tables trigger page faults which would be captured by Xen's hypervisor. Xen checks the PTE access right of the guest OS and sets PTE in shadow page tables to writable if the guest OS is writable to the PTE. Then we can record the updates in shadow page tables into a dirty bitmap.

By this way, we will be able to capture the occurrence of dirty pages and obtain a dirty page bitmap. Xen provides an API function `xc_shadow_control()` to handle shadow page tables. This feature can be turned on by calling `xc_shadow_control()` and setting flag as `XEN_DOMCTL_SHADOW_OP_ENABLE_LOGDIRTY` before live migration and turned off by setting `XEN_DOMCTL_SHADOW_OP_OFF` flag after migration finished.

## 4. Experiment Results

In this section, we present an evaluation of three-stage copy on Xen 4.1.4 and compare it with Xen's original precopy approach.

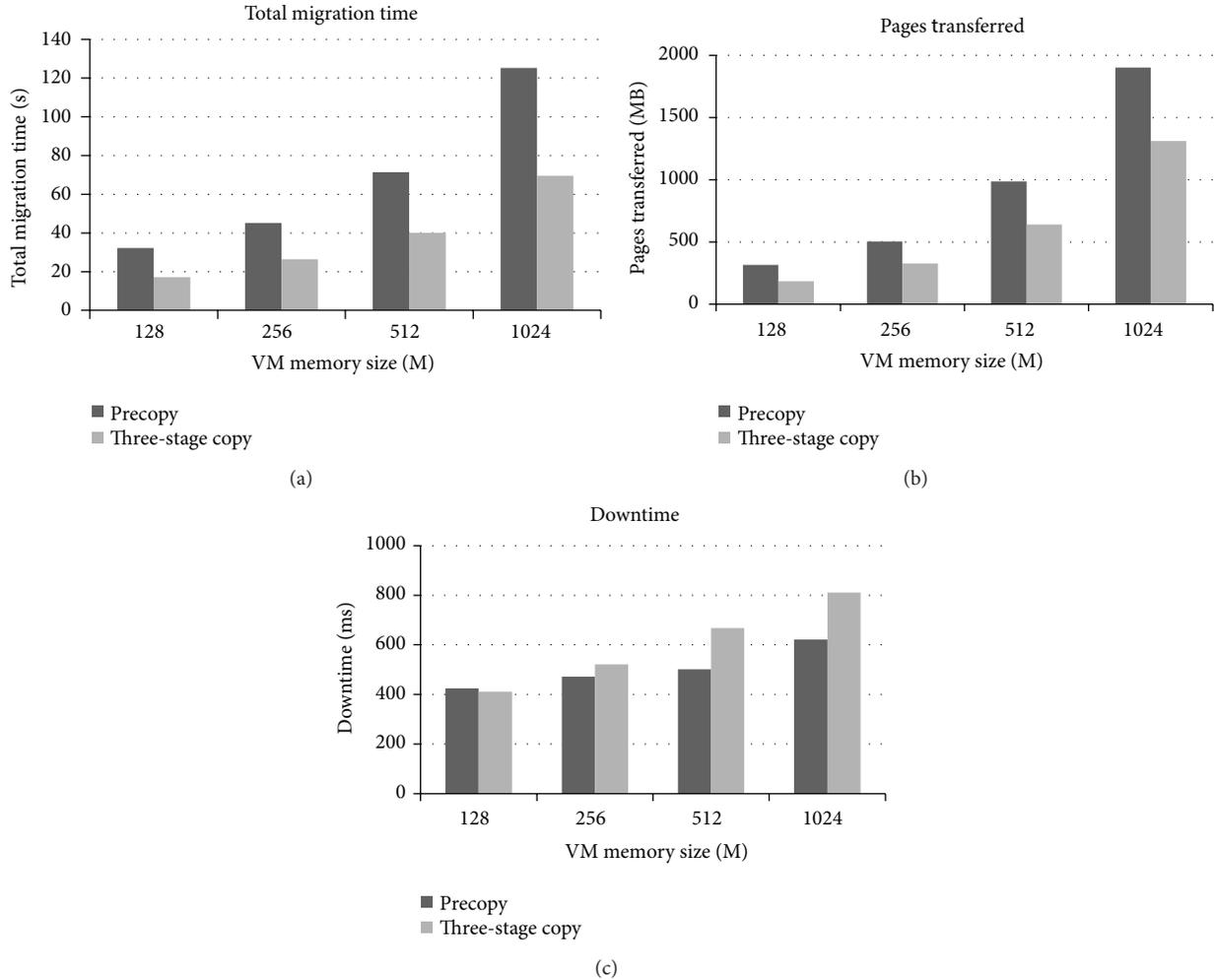


FIGURE 5: Comparison of total migration time, pages transferred, and downtime.

We conduct our experiments on two identical server-class machines, each with 2-way quad-core Xeon E5506 2.13 GHz CPUs and 32 GB DDR RAM, connected via a Gigabit Ethernet switch. All VM images are stored in a NFS server. We use Ubuntu 12.04 (Linux version 3.5.0-23) as guest OS and the privileged domain OS (domain 0). The host kernel is the modified version of Xen 4.1.4. Both the VM in each experiment and the domain 0 are configured to use two VCPUs. Guest VM sizes range from 128 MB to 1024 MB. And we use memtester [24] in virtual machine to generate high memory usage.

Each experiment is repeated five times and every test result comes from the arithmetic average of five values. In migration process, we evaluate three primary metrics discussed in Section 3: downtime, total migration time, and page transferred.

**4.1. Low Dirty Pages Rate.** Figure 4(a) shows that three-stage copy significantly reduces the total migration time for diverse VM memory size compared with precopy. With memory size increasing, the total migration time is reduced more. It reduces total migration time by average of 36.2%. In clusters

or data centers, less total migration time of VMs would get higher flexibility.

Figure 4(b) shows that three-stage copy approach also has the advantage in pages transferred; this should be attributed to less data transferred and lower network bandwidth needed. Experimental results show that the three-stage copy reduces pages transferred by average of 22%.

Evaluation in downtime Figure 4(c) shows that precopy could get stable downtime and three-stage copy's downtime would increase along with the increase of memory size. At low memory environment, three-stage copy needs less downtime than precopy, but it needs more downtime in large memory environment. It is due to the three-stage copy need to transfer dirty bitmap in suspend phase that large memory size would have more dirty pages. Nevertheless, the tradeoff between total migration time and downtime may be acceptable.

**4.2. High Dirty Pages Rate.** Figure 5(a) shows that in the case of high dirty pages rate the total migration time increases obviously in precopy while three-stage copy still maintains a shorter migration time. The three-stage copy reduces total migration time by average of 44.1% compared with precopy.

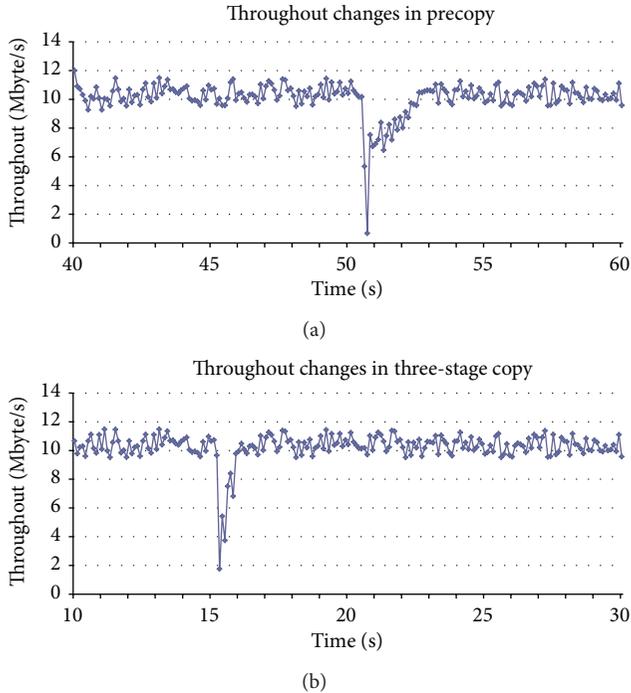


FIGURE 6: Comparison of total migration time, pages transferred, and downtime.

Figure 5(b) shows that three-stage copy approach still has the advantage in pages transferred over the precopy in the case of high dirty pages rate. As shown in the figure, it reduces pages transferred by average of 35.7%, which has a better optimization effect than in the case of low dirty pages rate.

Evaluation in Figure 5(c) shows that both precopy and three-stage copy have a large increase in downtime. This is because more dirty pages or dirty bitmap need to be transferred in downtime. In this test, longer downtime is needed in three-stage copy, but it still remains below 1 second in the worst case.

**4.3. Network IO.** We focus on the network throughput in the network IO test. As shown in Figure 5, the throughputs of both precopy and three-stage copy stay stable over a period of time and then suddenly drop and recover soon. The time when throughputs suddenly drop is the stage when the VMs down and copy pages. We can see that the duration of throughputs volatility is shorter in the three-stage copy. It means that throughput recovers faster from the bottom. It is because three-stage copy speeds up the transmission by the means of active push and prefetch pages after the VMs resume. As shown in Figure 6, it costs about 3 seconds to recover to normal throughput in precopy while only less than 1 second in three-stage copy.

## 5. Conclusions

This paper presents a three-stage memory copy (TSMC) approach for live virtual machine migration. In TSMC approach, the entire memory copy is divided into three stages: full memory copy, dirty bitmap copy, and dirty pages copy.

Most of the memory pages are just copied once; only dirtied pages need to be copied twice. It can significantly reduce the total pages transferred and cumulative migration time. Furthermore, because the TSMC approach just transfers dirty bitmap in stop phase of virtual machine, downtime is also shortened. Experimental results show that the TSMC approach could get better performance than Xen's precopy.

## Conflict of Interests

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

## References

- [1] T. Ohkawara, A. Aikebaier, T. Enokido, and M. Takizawa, "Quorums-based replication of multimedia objects in distributed systems," *Human-Centric Computing and Information Sciences*, vol. 2, article 11, 2012.
- [2] S. Silas, K. Ezra, and E. B. Rajsingh, "A novel fault tolerant service selection framework for pervasive computing," *Human-Centric Computing and Information Sciences*, vol. 2, article 5, 2012.
- [3] B. Meroufel and G. Belalem, "Dynamic replication based on availability and popularity in the presence of failures," *The Journal of Information Processing Systems*, vol. 8, no. 2, pp. 263–278, 2012.
- [4] N. Y. Yen and S. Y. F. Kuo, "An intergrated approach for internet resources mining and searching," *The KITCS/FTRA Journal of Convergence*, vol. 3, no. 3, pp. 37–44, 2012.
- [5] F. Xhafa, "Processing and analysing large log data files of a virtual campus," *The KITCS/FTRA Journal of Convergence*, vol. 3, no. 2, pp. 1–8, 2012.
- [6] Y. Pan and J. Zhang, "Parallel programming on cloud computing platforms—challenges and solutions," *The KITCS/FTRA Journal of Convergence*, vol. 3, no. 4, pp. 23–28, 2012.
- [7] E.-H. Song, H.-W. Kim, and Y.-S. Jeong, "Visual monitoring system of multi-hosts behavior for trustworthiness with mobile cloud," *The Journal of Information Processing Systems*, vol. 8, no. 2, pp. 347–358, 2012.
- [8] B. J. Oommen, A. Yazidi, and O.-C. Granmo, "An adaptive workflow scheduling scheme based on an estimated data processing rate for next generation sequencing in cloud computing," *The Journal of Information Processing Systems*, vol. 8, no. 4, pp. 191–212, 2012.
- [9] C. Waldspurger, "Memory resource management in VMware ESX server," in *ACM Operating Systems Design and Implementation*, pp. 181–194, VMware, 2002.
- [10] R. P. Goldberg, "Survey of virtual machine research," *IEEE Computer*, pp. 34–45, 1974.
- [11] G. H. S. Carvalho, I. Woungang, A. Anpalagan, and S. K. Dhurandher, "Virtual machine history model framework for a data cloud digital investigation," *The KITCS/FTRA Journal of Convergence*, vol. 3, no. 4, pp. 15–22, 2012.
- [12] T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, "Black-box and gray-box strategies for virtual machine migration," in *Proceedings of the 4th USENIX Symposium on Networked Systems Design and Implementation*, pp. 229–242, 2007.
- [13] D. Kapil, E. S. Pilli, and R. C. Joshi, "Live virtual machine migration techniques: survey and research challenges," in *Proceedings*

- of the Advance Computing Conference (IACC '13), pp. 963–969, 2013.
- [14] P. G. J. Leelipushpam and J. Sharmila, “Live VM migration techniques in cloud environment—a survey,” in *Proceedings of the Information & Communication Technologies (ICT '13)*, pp. 408–413, 2013.
- [15] X. Xie, H. Jiang, H. Jin, W. Cao, P. Yuan, and L. Yang, “Metis: a profiling toolkit based on the virtualization of hardware performance counters,” *Human-Centric Computing and Information Sciences*, vol. 2, article 8, 2012.
- [16] P. Barham, B. Dragovic, K. Fraser et al., “Xen and the art of virtualization,” in *Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP '03)*, pp. 164–177, October 2003.
- [17] C. Clark, K. Fraser, S. Hand et al., “Live migration of virtual machines,” in *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation (NSDI '05)*, pp. 273–286, 2005.
- [18] A. Kivity, Y. Kamay, and D. Laor, “KVM: the linux virtual machine monitor,” in *Proceedings of the Ottawa Linux Symposium*, pp. 225–230, 2007.
- [19] M. Nelson, B. Lim, and G. Hutchines, “Fast transparent migration for virtual machines,” in *Proceedings of the USENIX Annual Technical Conference*, pp. 391–394, 2005.
- [20] M. R. Hines and K. Gopalan, “Post-copy based live virtual machine migration using pre-paging and dynamic self-ballooning,” in *Proceedings of the ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE '09)*, pp. 51–60, March 2009.
- [21] H. Jin, D. Li, S. Wu, X. Shi, and X. Pan, “Live virtual machine migration with adaptive memory compression,” in *Proceedings of the IEEE International Conference on Cluster Computing and Workshops (CLUSTER '09)*, September 2009.
- [22] L. Haikun, J. Hai, L. Xiaofei, H. Liting, and Y. Chen, “Live migration of virtual machine based on full system trace and replay,” in *Proceedings of the 18th ACM International Symposium on High Performance Distributed Computing (HPDC '09)*, pp. 101–110, June 2009.
- [23] H. A. Lagar-Cavilla, J. A. Whitney, R. Bryant et al., “SnowFlock: Virtual machine cloning as a first-class cloud primitive,” *ACM Transactions on Computer Systems*, vol. 29, no. 1, article 2, 2011.
- [24] A utility for testing memory, <http://pyropus.ca/software/memtester/>.

## Research Article

# Grid-PPPS: A Skyline Method for Efficiently Handling Top- $k$ Queries in Internet of Things

Sun-Young Ihm,<sup>1</sup> Aziz Nasridinov,<sup>2</sup> and Young-Ho Park<sup>1</sup>

<sup>1</sup> Department of Multimedia Sciences, Sookmyung Women's University, Cheongpa-ro 47-gil 100, Yongsan-gu, Seoul 140-742, Republic of Korea

<sup>2</sup> School of Computer Engineering, Dongguk University at Gyeongju, 123 Dongdae-ro, Gyeongju, Gyeongbuk 780-714, Republic of Korea

Correspondence should be addressed to Young-Ho Park; yhpark@sm.ac.kr

Received 22 January 2014; Accepted 7 April 2014; Published 8 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Sun-Young Ihm 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 rapid development in wireless communication and radio frequency technology has enabled the Internet of Things (IoT) to enter every aspect of our life. However, as more and more sensors get connected to the Internet, they generate huge amounts of data. Thus, widespread deployment of IoT requires development of solutions for analyzing the potentially huge amounts of data they generate. A top- $k$  query processing can be applied to facilitate this task. The top- $k$  queries retrieve  $k$  tuples with the lowest or the highest scores among all of the tuples in the database. There are many methods to answer top- $k$  queries, where skyline methods are efficient when considering all attribute values of tuples. The representative skyline methods are soft-filter-skyline (SFS) algorithm, angle-based space partitioning (ABSP), and plane-project-parallel-skyline (PPPS). Among them, PPPS improves ABSP by partitioning data space into a number of spaces using hyperplane projection. However, PPPS has a high index building time in high-dimensional databases. In this paper, we propose a new skyline method (called Grid-PPPS) for efficiently handling top- $k$  queries in IoT applications. The proposed method first performs grid-based partitioning on data space and then partitions it once again using hyperplane projection. Experimental results show that our method improves the index building time compared to the existing state-of-the-art methods.

## 1. Introduction

A rapid development in wireless communication and radio frequency technology has enabled the Internet of Things (IoT) to enter every aspect of our life. The IoT is part of the internet of the future and will comprise billions of intelligent communicating “things” which will have sensing, actuating, and data processing capabilities [1]. For example, the things in IoT can be smart devices in home or home appliances such as refrigerator, washing machine, and air conditioner, which have controllable devices. Restaurants, hotels, and countries can be also considered as the things in IoT, since they are connected and communicate with each other. However, as more and more sensors get connected to the Internet, they generate enormous amounts of data. Thus, widespread deployment of IoT requires development of solutions for analyzing the potentially huge amounts of data they generate

[2–4]. A top- $k$  query processing can be applied to facilitate this task.

The top- $k$  query finds  $k$  tuples with the lowest or the highest scores among all of the input tuples. When a database is large, it may take long computing time to find a complete answer to a query. Most users, however, are interested in looking at just a few top results, which are ranked by a small set of attribute values, and they want to see the results immediately after they issue the query [5]. We can apply this notion to find the top- $k$  results in huge amounts of data in IoT applications. Example 1 presents the scenario to find the top- $k$  results in IoT applications.

*Example 1.* Consider a user John, who wants to have a dinner in an Italian restaurant. He defines the following criteria for the search: the distance of restaurant from his home should be less than 800 meters and price should be less than

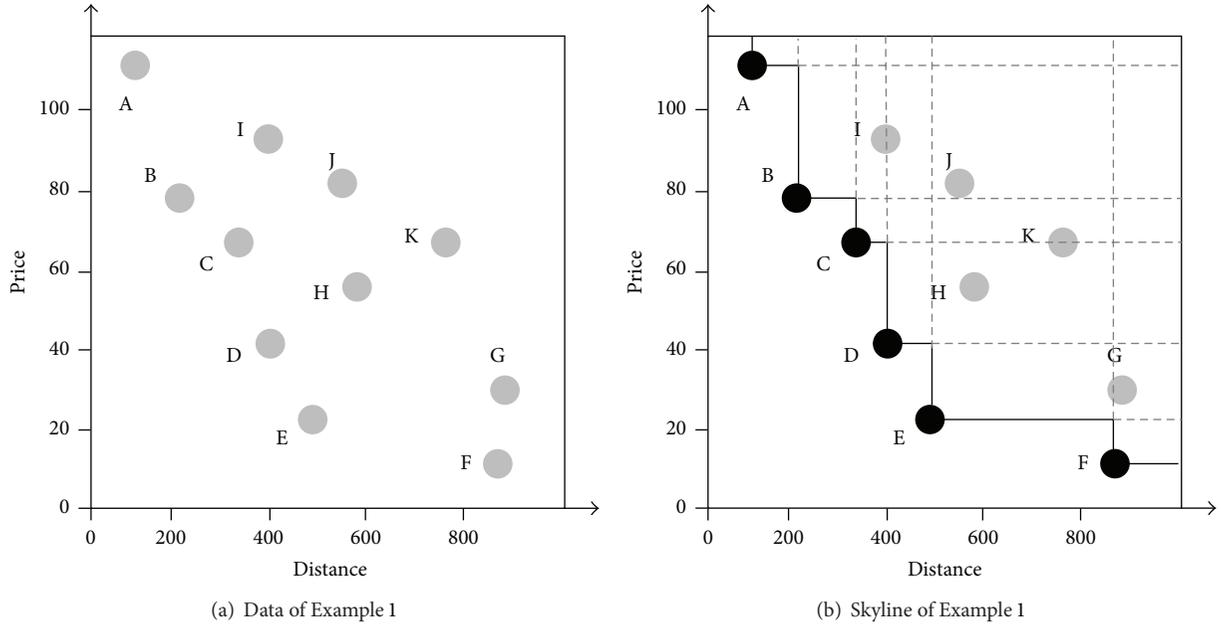


FIGURE 1: Graphical representation of data and skyline.

TABLE 1: The list of restaurants in Example 1.

| Number | Name         | Distance (100 m) | Price (\$10) | Score    |
|--------|--------------|------------------|--------------|----------|
| A      | Mago         | 1.0              | 11.0         | No score |
| B      | Little Pasta | 2.0              | 8.0          | 5.6      |
| C      | La Tavola    | 3.5              | 7.0          | 5.6      |
| D      | Olive Garden | 4.0              | 4.0          | 4.0      |
| E      | Alto         | 5.0              | 2.0          | 3.2      |
| F      | Melrose      | 8.8              | 1.0          | No score |
| G      | Morton House | 9.0              | 3.0          | No score |
| H      | Applebee's   | 6.0              | 6.0          | 6.0      |
| I      | Boulevard    | 4.0              | 9.0          | No score |
| J      | IHOP         | 5.3              | 8.5          | No score |
| K      | Shadow Brook | 7.6              | 7.0          | 7.24     |

85 US Dollars. In order to find a restaurant that best suits his interest, John makes a scoring function  $f$  as  $f(t) = 0.4 * \text{distance} + 0.6 * \text{price}$ , where  $t$  is the tuple of database. Here, we can think of the restaurant as a thing in IoT. All restaurants are connected to the Internet, which forms the network of IoT. Finding the top- $k$  results among a large amount of restaurants could save John's time. The query shown below is based on the PostgreSQL syntax.

```
SELECT *
FROM Restaurant R
WHERE R.distance < 8.0 AND R.price < 8.5
ORDER BY f(t)
```

The list of restaurants and their scores are shown in Table 1. These restaurants can be represented in two-dimensional space as shown in Figure 1(a). The Alto, E, is top-1 answer to the query with a score of 3.2 and Olive Garden, D, is top-2 answer to the query with a score of 4.0. Since restaurants

A, F, G, I, and J have higher values for distance and price, they do not satisfy the requirements provided by John. Thus, the scores for these restaurants are not calculated.

To answer the top- $k$  queries efficiently, building an index by accessing the subset of database is needed. The skyline methods are representative methods for answering the top- $k$  queries by constructing skyline as an index. These methods express data tuples as objects in a  $d$ -dimensional space and then construct a skyline. Here,  $d$  is the number of attributes of a database. The skyline methods are efficient for queries in a database with a large number of attributes and data. In Figure 1(b), the rectangular black line, composed of black points, represents the skyline. The skyline points do not dominate each other. We can answer top- $k$  queries only by reading the skyline points, since the skyline can be considered as an index. The soft-filter-skyline (SFS) algorithm [6], which is the state-of-the-art method, presorts the objects by calculating entropy value of object. The angle-based space partitioning (ABSP) [7] and plane-project-parallel-skyline (PPPS) [8] partition data space into a number of subregions in order to reduce the computing time. PPPS improves ABSP by partitioning data space into a number of spaces using hyperplane projection. However, PPPS has a high index building time in high-dimensional databases.

In this paper, we propose a new skyline method for efficiently handling top- $k$  queries in IoT applications. This paper focuses on the effectiveness of grid-based partitioning. More precisely, the contributions we make in this paper are as follows.

- (i) We propose a new skyline method (called Grid-PPPS) for efficiently handling top- $k$  queries in IoT applications. The proposed method first performs grid-based partitioning on data space and then partitions it once

again using hyperplane projection. This reduces the time complexity of the PPPS.

- (ii) We show the performance advantages of the Grid-PPPS through the comparison of the index building time and number of dominating objects compared to PPPS.

The rest of this paper is organized as follows. Section 2 describes existing work related to this paper. Section 3 presents the proposed method for computing Grid-PPPS and Section 4 demonstrates the results of performance evaluation. Section 5 summarizes and concludes the paper.

## 2. Related Work

In this section, we discuss the existing literature. In Section 2.1, we review data management solutions in IoT, and, in Section 2.2, we explain the index building methods for top- $k$  queries.

*2.1. Data Management Methods in IoT.* Generally speaking, all things on the IoT may generate a huge amount of data that contains different kinds of useful information. However, how to handle such big data and how to retrieve the valuable information have become hot research topic in recent years. Several index building methods for handling massive amount of IoT data are proposed. Ma et al. [9] proposed an update and query efficient index framework (UQE-Index) based on key-value store that can support both multidimensional query and high insert throughput. In order to effectively reduce the index update times and decrease the index maintenance cost, the authors proposed a dynamic data partition strategy that can make sure that the data is evenly distributed into each region in HBase and the data that is close in time and space dimension is usually stored in the same regions.

In order to address the problem of high dimensionality in IoT data, Huang et al. [10] proposed dynamic skyline cube (SKYCUBE) computation to efficiently balance the computation update and costs in IoT. The authors proposed an efficient grid-based ADSCIT (algorithm for dynamic SKYCUBE computation in the Internet of Things) which consists of two modules: continuous maintenance module (CMM), which incrementally updates the nonpseudo objects, and progressive computation module (PCM), which can rapidly obtain the skyline cube from the updated nonpseudo objects. In order to integrate the proposed two modules, a grid-based evaluation method that uses regular grid index is proposed.

Elkheir et al. [11] surveyed the data management solutions that are proposed for IoT and proposed a data management framework that takes into consideration the drawbacks of existing approaches. The proposed framework adapts a federated, data and sources centric approach to link diverse things with their abundance of data to the potential applications and services. Data mining technologies can also be used to discover the hidden information in the data of IoT, which can be used to improve the performance of the system or to enhance quality of services this new environment can

provide [12]. Tsai et al. [12] surveyed research on how to connect data mining technologies to the IoT, which include clustering, classification, and frequent patterns mining technologies, from a different perspective. The authors also discuss changes, potentials, open issues, and future trends of applying data mining to the IoT.

*2.2. Index Building Methods for Top- $k$  Queries.* To construct an index efficiently, skyline and convex hull methods are representative methods. These methods construct an index as a list of layers and consist of objects which are not dominated by each other. The computing cost of skyline methods is much lower than that of convex hull methods; however, the number of objects in each layer of skyline methods is much larger than that of convex hull methods. Thus, the skyline methods are mainly used in the applications where insertion, update, and deletion operations are frequently occurring on objects. Since such applications need to construct skyline more frequently, they require small computing time. On the other hand, objects in convex hull methods are not updated often. Thus, these methods are used in the applications where top- $k$  query processing is performed. This is because a layer in convex hull methods consists of small number objects, which results in rapid processing of top- $k$  queries. In this paper, we focus on reducing the index construction of skyline in which data is frequently updated.

*2.2.1. Skyline Methods.* The skyline methods are useful when answering top- $k$  queries by accessing only a subset of the database. These methods have an advantage of low index building cost. The skyline operation was first introduced by Köhler et al. [8] and there have been a number of variations of it. The data space partitioning technique is used in many skyline methods for early pruning objects which are not included in skyline. There are several algorithms for constructing skyline that apply space partitioning technique. Grid-based data space partitioning has been commonly used in distributed and parallel skyline processing [8]. The angle-based space partitioning approach (ABSP) [7] is proposed by using hyperspherical coordinates of data objects and improves grid-based space partitioning. Köhler et al. [8] proposed a novel approach called PPPS, which reduces the computing time of ABSP by coordinating the objects using hyperplane projection.

There are also other algorithms for constructing skyline and the representative methods are block nested loops (BNL) [13], SFS [6], and linear elimination sort for skyline (LESS) [14]. BNL sequentially reads the input relation and saves in a window  $w$ . When an object  $o$  is read, it is compared to objects in  $w$ . If an object in  $w$  dominates  $o$ , BNL eliminates  $o$ . Otherwise,  $o$  dominates some objects in  $w$ ; these are deleted from  $w$  and  $o$  is added to [13]. The SFS algorithm [6] improves BNL by presorting the input relation according to the entropy value of object. LESS is an improvement of SFS that essentially combines aspects of a number of the established algorithms [14]. LESS discards some dominating objects earlier; thus this has the advantage of reducing the number of pairwise comparisons between the objects than

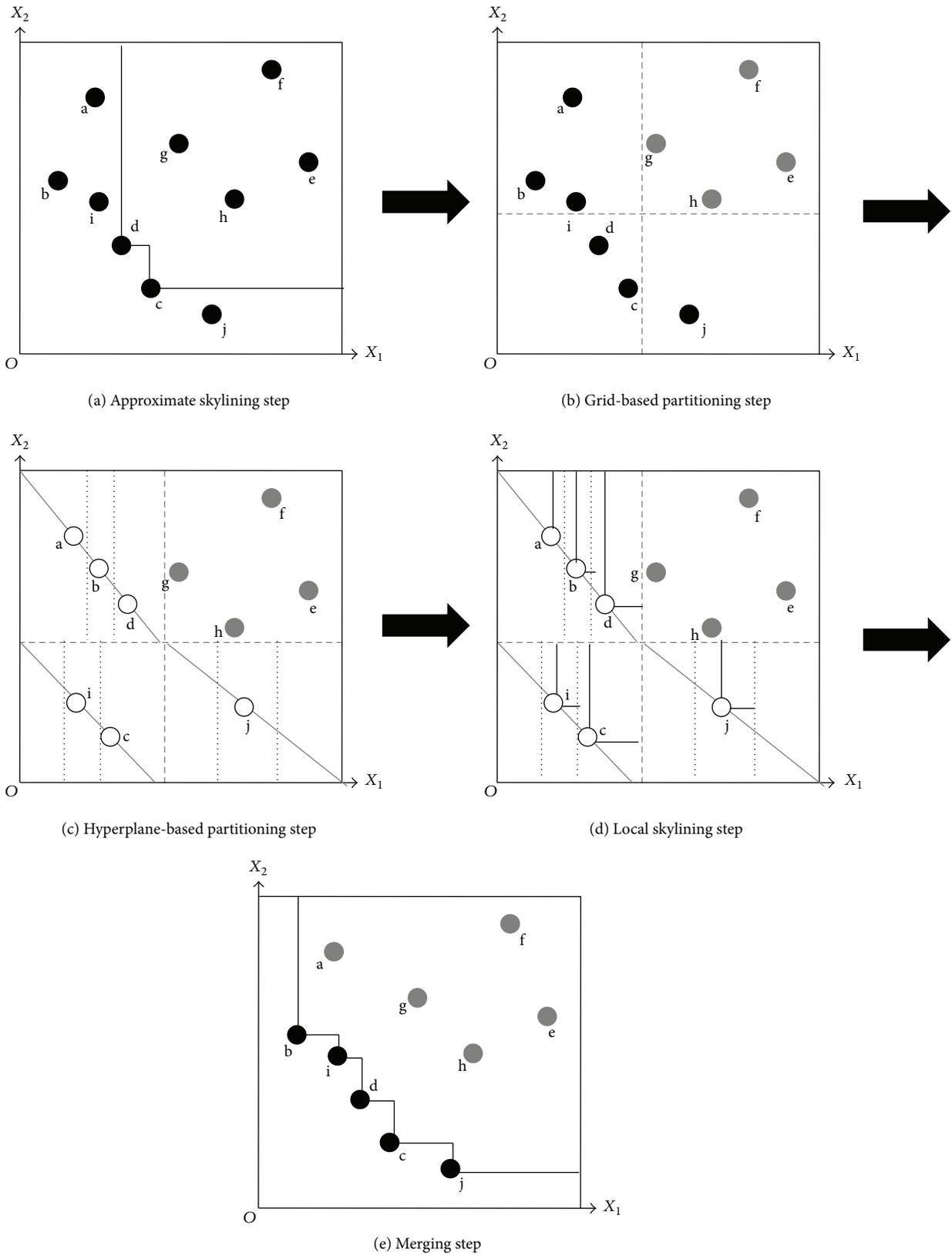


FIGURE 2: The overall procedure for processing Grid-PPPS in the two-dimensional data space.

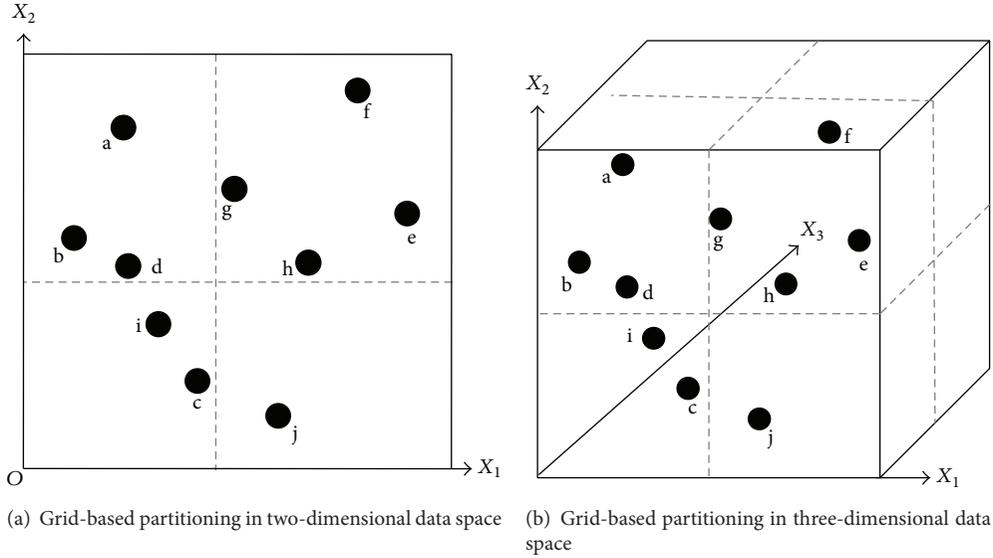


FIGURE 3: The example of grid-based partitioning.

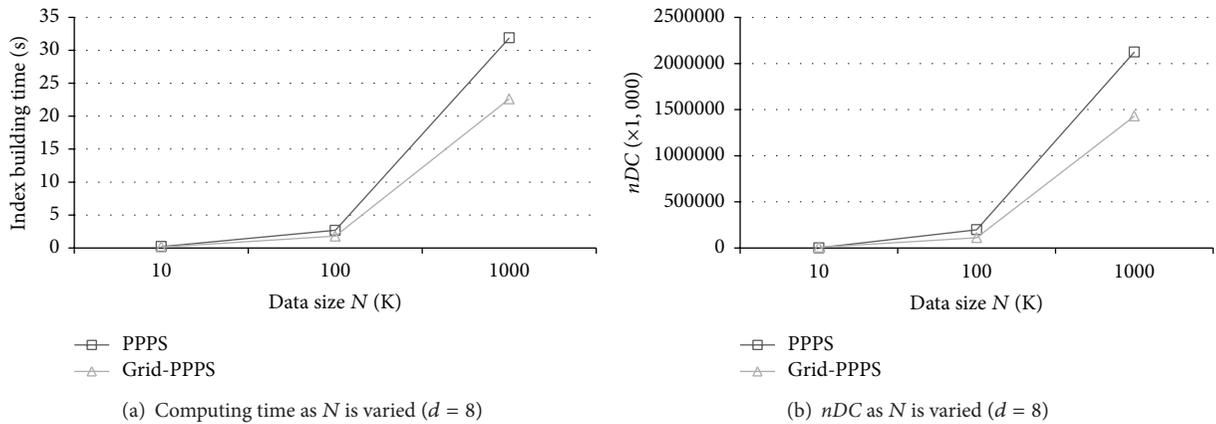


FIGURE 4: The comparison of the computing time and  $nDC$  as  $N$  is varied related to Experiment 1.

SFS. However, the number of comparisons is still large. There also has been a growing interest in distributed [15, 16] and parallel [17, 18] skyline computation lately.

**2.2.2. Other Methods.** The convex hull methods construct the layer of edge objects in a convex hull shape and discard other objects. The layer size of convex hull methods is smaller than that of skyline methods; however, the index building time of convex hull methods is higher than that of skyline methods. The representative convex hull methods are ONION [19] and HL-Index [5]. ONION [19] builds convex hull as an index by constructing a boundary with the edge objects. That is, the objects of the first layer encircle the other objects. ONION builds a second layer in the same manner and finally constructs a list of layers as a result. HL-Index [5] builds a convex hull as ONION does and sorts lists additionally for retrieving top- $k$  results efficiently.

In order to reduce the index building time of convex hull methods, there are some methods that combine convex hull and skyline methods. For example, Ihm et al. [20] proposed the approximate convex skyline (AppCS) method that constructs skyline over the entire objects and then partitions it. Further, AppCS builds an approximate convex hull in each partitioned region with virtual objects. Another method that focuses on reducing index building time of convex hull is proposed in [21]. The authors proposed a method called approximate convex hull index (aCH-Index) that computes the skyline over the entire set of objects, partitions the region into multiple subregions to reduce the computing time of convex hull in all origins, and then computes the convex hull in each subregion.

### 3. Grid-PPPS

In this section, we explain the proposed methods, Grid-PPPS. As explained in Section 2.1, the PPS [8] improves

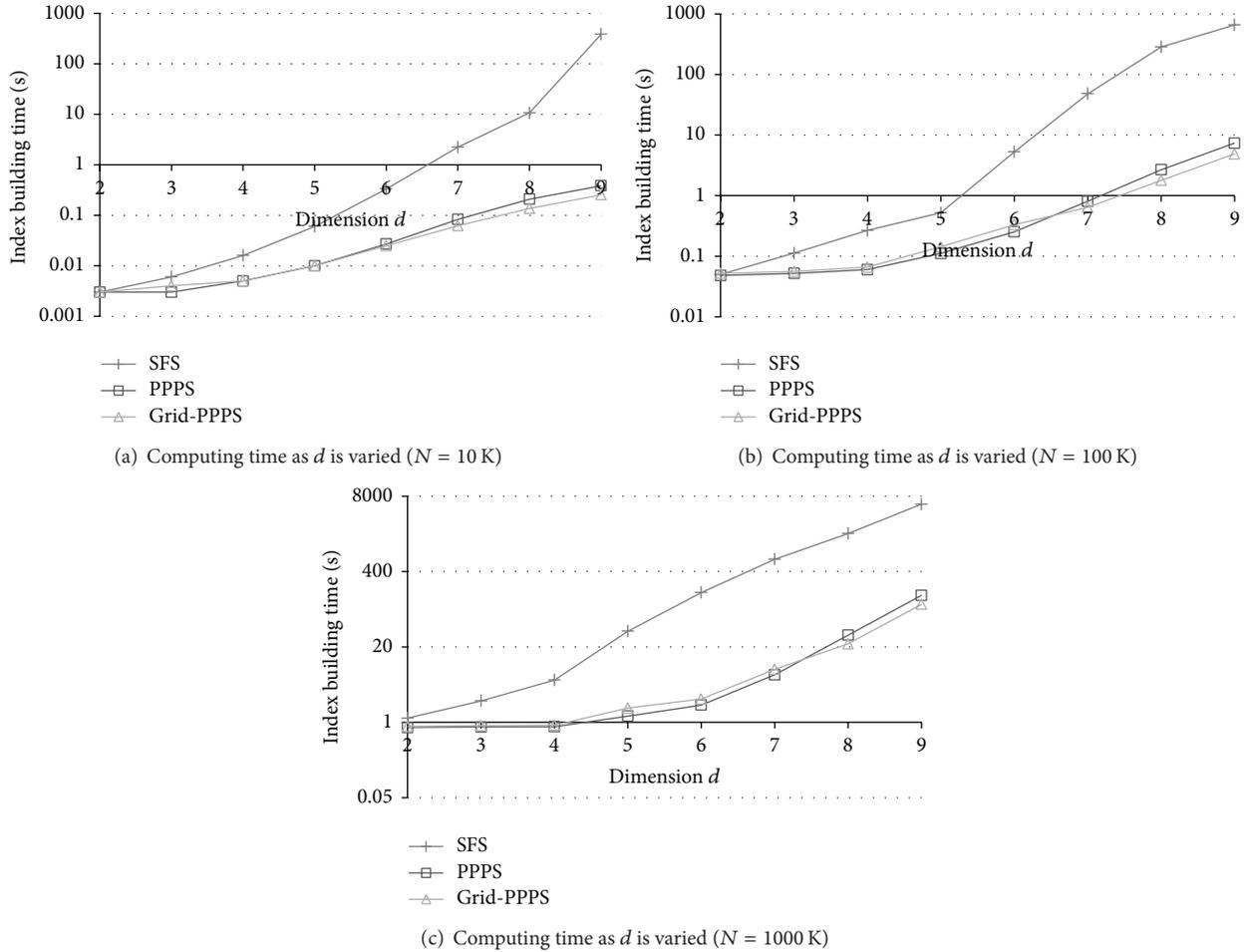


FIGURE 5: The comparison of the computing time of Grid-PPS and SFS as  $d$  and  $N$  are varied related to Experiment 2.

the indexing building time of ABSP [7]. However, PPS has a high index building time in high-dimensional databases. The Grid-PPS reduces the time complexity of the PPS. The Grid-PPS is constructed by five steps as shown in Figure 2: (a) approximate skylining step, (b) grid-based partitioning step, (c) hyperplane-based partitioning step, (d) local skylining step, and (e) merging step. For the convenience of the explanation, Figure 2 shows the procedure of processing Grid-PPS in two-dimensional region. We explain each step in detail from Sections 3.1 to 3.5.

**3.1. Approximate Skylining Step.** In the first step, Grid-PPS constructs approximate skyline. This step is shown in Figure 2(a). Computing the exact skyline of all tuples set  $T$  can be expensive, since each tuple should be compared to many other tuples. However, we can prune several tuples with the few comparisons. We prune the objects by calculating the entropy value of each object. We select several tuples, which have low entropy value, and then make a small set  $S \subset T$  with those tuples. By the small set  $S$ , some tuples in  $T$  are dominated by  $S$ , and those tuples can be eliminated safely. Since we pick the tuples according to entropy value, we can

discard more tuples. Finally, we can get approximate skyline. Importantly, for fixed size  $S$ , computing the approximate skyline can be performed in a linear time with a single pass over the dataset [8].

**3.2. Grid-Based Partitioning Step.** In the major step that is grid-based partitioning step, Grid-PPS partitions the data space into  $b$  subspaces using grid-based partitioning technique. A grid is something which is in a pattern of straight lines that cross over each other, forming squares. Many applications are using grid-base technique, since it is simple and has low computing cost [22–24]. The grid-based partitioning scheme is based on recursively dividing some dimension of the data space into two parts [7]. The computing time of grid-based partitioning is lower than other partitioning techniques, because grid-based partitioning is simple and cheap to compute. Thus, we partition objects, which are obtained from approximate skylining step into  $b$  spaces with grid-based partitioning technique. Figure 3(a) shows the example of grid-based partitioning in two-dimensional data space, and three-dimensional example is shown in Figure 3(b).

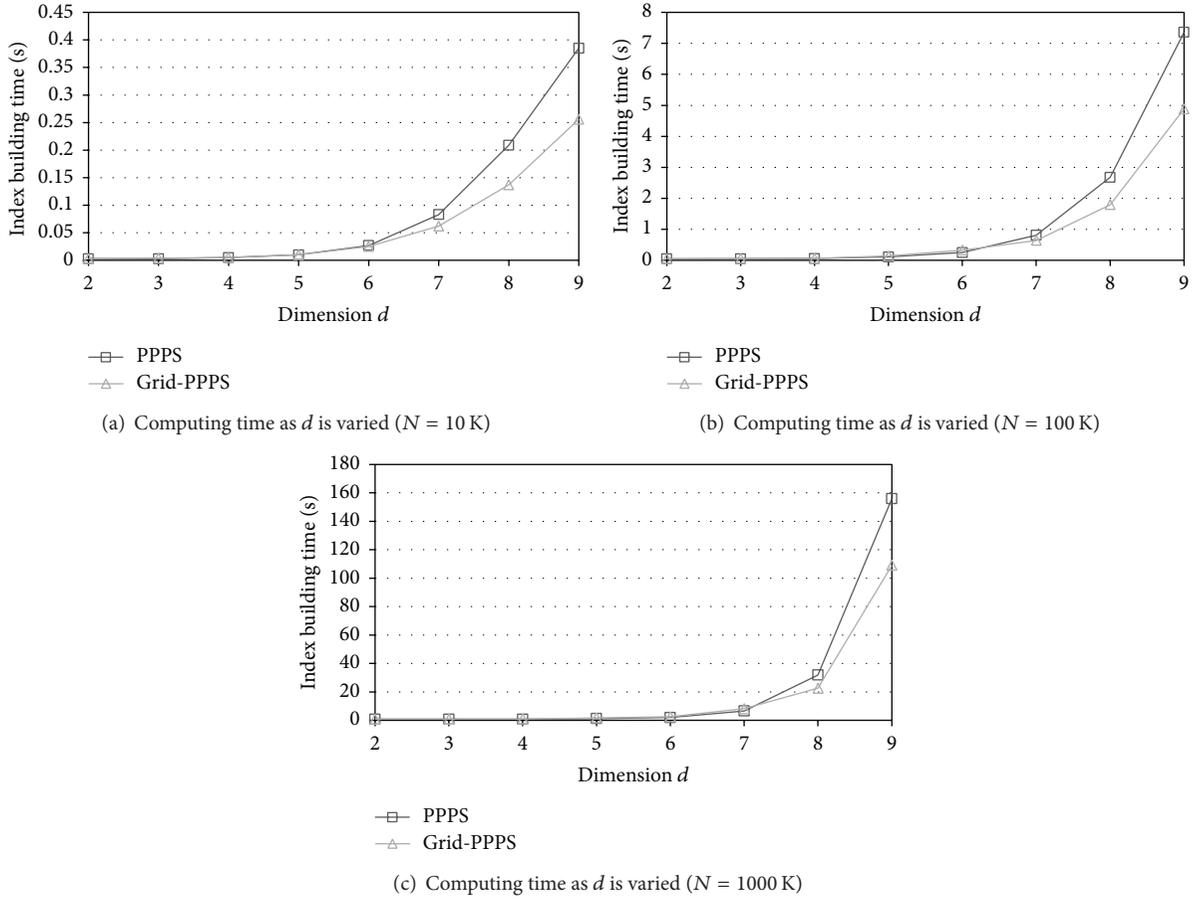


FIGURE 6: The comparison of the computing time of the Grid-PPPS and PPPS as  $d$  and  $N$  are varied related to Experiment 2.

**3.3. Hyperplane-Based Partitioning Step.** In the hyperplane-based partitioning step, Grid-PPPS partitions space into  $c$  subspaces using hyperplane-based partitioning, which is proposed in PPPS [8]. We first calculate the formula of the hyperplane such as  $x_1 + x_2 + \dots + x_d = 1$ . Next, we project tuples onto the hyperplane, and (1) shows the calculation of projection. Finally we partition space, which consists of projected tuples, into  $c$  subspaces:

$$(x_1 \dots x_d) \mapsto (x_1 \dots x_d) \times \frac{1}{x_1 + \dots + x_d}. \quad (1)$$

**3.4. Local Skylining Step.** In the local skylining step, Grid-PPPS computes the local skyline in each subspace <sub>$ij$</sub> . We call local skyline in subspace <sub>$i$</sub>  as subskyline <sub>$i$</sub>  and use SFS algorithm [6] for computing subskyline. For the construction of local skyline, the dominating calculation, which determines whether the object is in the skyline or not, should be computed between two objects. Grid-PPPS filters out the objects by grid-based partitioning step, and, thus, the number of dominating calculation decreases.

**3.5. Merging Step.** In the last step, Grid-PPPS combines the subskylines in each subspace. We build a layer by merging the subskylines. Since Grid-PPPS computes subskyline points

once again, it combines the subskylines and builds a result layer without losing tuples and overlapping.

## 4. Performance Evaluation

In this section, we first explain the data and environment in Section 4.1 and then present the results of experiments in Section 4.2.

**4.1. Experimental Data and Environment.** We have implemented the proposed method using C++. We conduct all the experiments on an Intel i5-760 quad core processor running at 2.80 GHz Linux PC with 16 GB of main memory. We use the uniform dataset for all of our experiment data. We use 10 K, 100 K, and 1000 K data size. We experiment our data in two through nine dimensions.

**4.2. Result of Experiments.** We compare the computing time and the  $nDC$  (number of domination calculation) of the Grid-PPPS with the existing methods PPPS [8] and SFS [6]. We use the wall clock time as the measure of the computing time. We measure the computing time  $nDC$  on the synthetic dataset while varying the data size  $N$  and the dimension  $d$ .

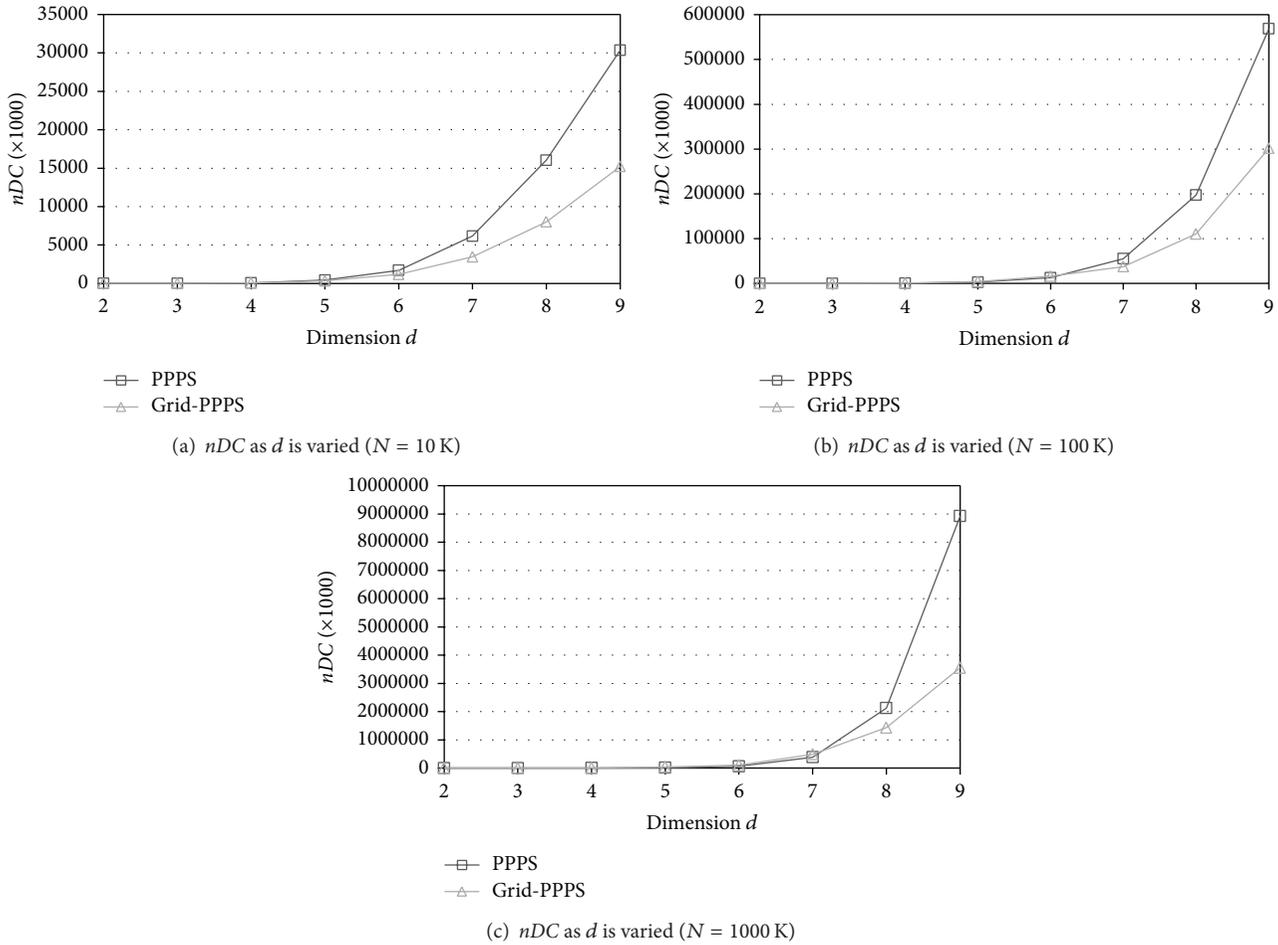


FIGURE 7: The comparison of the  $nDC$  and  $d$  and  $N$  are varied related to Experiment 3.

The result of the skyline constructed by Grid-PPPS is exactly the same as PPPS. Grid-PPPS improves the index building time of PPPS in large and high-dimensional dataset. When data has 10 K size and under six attributes, the index building time of Grid-PPPS is a little higher than PPPS, because of partitioning step. The number of filtered tuples in Grid-PPPS is similar to PPPS in the small and low-dimensional dataset. However, Grid-PPPS constructs an index much quickly in large and high-dimensional dataset as shown in experiments.

*Experiment 1.* Computing time and  $nDC$  as data size  $N$  is varied.

Figure 4(a) shows the computing time of Grid-PPPS and PPPS as  $N$  is varied from 10 K to 1000 K. The result increases in log scale as shown in Figure 4. The computing time of the Grid-PPPS improves by 1.41–1.52 times over the PPPS. Figure 4(b) shows the  $nDC$  of Grid-PPPS and PPPS as  $N$  is varied from 10 K to 1000 K. The  $nDC$  of Grid-PPPS improves 1.49–2.00 times over the PPPS.

*Experiment 2.* Computing time as dimension  $d$  and data size  $N$  are varied.

Figures 5(a), 5(b), and 5(c) show the computing time of Grid-PPPS and PPPS as  $d$  is varied from 2 to 9 and  $N$  is varied from 10 K to 1000 K. The result increases in log scale as shown in Figure 5. Figure 5(a) shows the computing time of the Grid-PPPS improves by 0.75–1.52 times over the PPPS as  $d$  is varied and  $N$  is 10 K. Figure 5(b) shows the computing time of the Grid-PPPS improves by 0.77–1.51 times over the PPPS as  $d$  is varied and  $N$  is 100 K. Figure 5(c) shows the computing time of the Grid-PPPS improves by 0.73–1.43 times over the PPPS as  $d$  is varied and  $N$  is 1000 K. In order to show the precise difference between Grid-PPPS and PPPS, we conduct the experiments shown in Figure 6.

*Experiment 3.* The  $nDC$  as dimension  $d$  and data size  $N$  are varied.

Figures 7(a), 7(b), and 7(c) show the  $nDC$  of Grid-PPPS and PPPS as  $d$  is varied from 2 to 9 and  $N$  is varied from 10 K to 1000 K. The result increases in log scale as shown in Figure 7. Figure 7(a) shows the  $nDC$  of the Grid-PPPS improves by 1.00–2.01 times over the PPPS as  $d$  is varied and  $N$  is 10 K. Figure 7(b) shows the  $nDC$  of the Grid-PPPS improves by 0.68–1.89 times over the PPPS as  $d$  is varied and  $N$  is 100 K. Figure 7(c) shows the  $nDC$  of the Grid-PPPS

improves by 0.65–1.49 times over the PPPS as  $d$  is varied and  $N$  is 1000  $K$ .

## 5. Conclusion

As more and more sensors get connected to the Internet, the IoT applications generate enormous amounts of data. In order to solve this problem, in this paper, we have proposed to use a top- $k$  query processing to find the best results among vast amount of data. In order to efficiently handle top- $k$  queries, we have proposed a new skyline method called Grid-PPPS, which performs grid-based partitioning first on data space and then partitions it once again using hyperplane projection. We have compared the proposed method with the state-of-the-art methods, such as PPPS and SFS. The results of experiments demonstrate several times improvement in most cases.

## Conflict of Interests

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

## Acknowledgment

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012003797).

## References

- [1] C. Perera, A. Zaslavsky, C. H. Liu, M. Compton, P. Christen, and D. Georgakopoulos, "Sensor search techniques for sensing as a service architecture for the internet of things," *IEEE Sensors Journal*, vol. 14, no. 2, pp. 406–420, 2014.
- [2] C. Zhu, Q. Zhu, C. Zuzarte, and W. Ma, "Developing a dynamic materialized view index for efficiently discovering usable views for progressive queries," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 511–537, 2013.
- [3] Y. Park, K. Whang, B. S. Lee, and W. Han, "Efficient evaluation of partial match queries for XML documents using information retrieval techniques," in *Proceedings of the Database Systems for Advanced Applications (DASFAA '05)*, pp. 95–112, April 2005.
- [4] R. M. Hwang, S. K. Kim, S. An, and D. W. Park, "The architectural pattern of a highly extensible system for the asynchronous processing of a large amount of data," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 511–537, 2013.
- [5] J. Heo, J. Cho, and K. Whang, "The hybrid-layer index: a synergic approach to answering Top- $k$  queries in arbitrary subspaces," in *Proceedings of the 26th international Conference on Data Engineering*, pp. 445–448, March 2010.
- [6] J. Chomicki, P. Godfrey, J. Gryz, and D. Liang, "Skyline with presorting," in *Proceedings of the 19th International Conference on Data Engineering*, pp. 717–719, March 2003.
- [7] A. Vlachou, C. Doukeridis, and Y. Kotidis, "Angle-based space partitioning for efficient parallel skyline computation," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 227–238, June 2008.
- [8] H. Köhler, J. Yang, and X. Zhou, "Efficient parallel skyline processing using hyperplane projections," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pp. 85–94, June 2011.
- [9] Y. Ma, J. Rao, W. Hu et al., "An efficient index for massive IOT data in cloud environment," in *Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM '12)*, pp. 2129–2133, 2012.
- [10] Z. Huang, Y. Xiang, D. Wang, and B. Zhang, "Efficient dynamic SKYCUBE computation in the internet of things," in *Proceedings of the International Conference on Computer and Communication Technologies in Agriculture Engineering (CCTAE '10)*, pp. 308–311, June 2010.
- [11] M. A. Elkheir, M. Hayajneh, and N. A. Ali, "Data management for the internet of things: design primitives and solution," *Sensors*, vol. 13, no. 11, pp. 15582–15612, 2013.
- [12] C. W. Tsai, C. F. Lai, M. C. Chiang, and L. T. Yang, "Data mining for internet of things: a survey," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 77–97, 2014.
- [13] S. Börzsönyi, D. Kossmann, and K. Stocker, "The skyline operator," in *Proceedings of the 17th International Conference on Data Engineering*, pp. 421–430, April 2001.
- [14] P. Godfrey, R. Shipley, and J. Gryz, "Maximal vector computation in large data sets," in *Proceedings of the 31st International Conference on Very Large Data Bases*, pp. 229–240, September 2005.
- [15] K. Hose, C. Lemke, and K.-U. Sattler, "Processing relaxed skylines in PDMS using distributed data summaries," in *Proceedings of the 2006 Conference on Information and Knowledge Management*, pp. 425–434, November 2006.
- [16] S. Wang, B. C. Ooi, A. K. H. Tung, and L. Xu, "Efficient skyline query processing on peer-to-peer networks," in *Proceedings of the 2007 International Conference on Data Engineering*, pp. 1126–1135, April 2007.
- [17] A. Cosgaya-Lozano, A. Rau-Chaplin, and N. Zeh, "Parallel computation of skyline queries," in *Proceedings of the 21st International Symposium on High Performance Computing Systems and Applications*, pp. 1–7, May 2007.
- [18] P. Wu, C. Zhang, Y. Feng, B. Y. Zhao, D. Agrawal, and A. E. Abbadi, "Parallelizing skyline queries for scalable distribution," in *Proceedings of the 2006 Conference on Extending Database Technology*, pp. 112–130, 2006.
- [19] Y. C. Chang, L. Bergman, V. Castelli, C. S. Li, M. L. Lo, and J. R. Smith, "The onion technique: indexing for linear optimization queries," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of data*, pp. 391–402, 2000.
- [20] S. Y. Ihm, K. E. Lee, A. Nasridinov, J. S. Heo, and Y. H. Park, "Approximate convex skyline: a partitioned layer-based index for efficient processing top- $k$  queries," *Knowledge-Based Systems*, vol. 61, pp. 13–28, 2014.
- [21] S. Y. Ihm, A. Nasridinov, and Y. H. Park, "An efficient index building algorithm for selection of aggregator node in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 520428, 8 pages, 2014.
- [22] J. W. K. Gnanaraj, K. Ezra, and E. B. Rajsingh, "Smart card based time efficient authentication scheme for global grid computing," *Human-Centric Computing and Information Sciences*, vol. 3, no. 16, pp. 1–14, 2013.

- [23] S. Hong and J. Chang, "A new  $k$ -NN query processing algorithm based on multicasting-based cell expansion in location-based services," *Journal of Convergence*, vol. 4, no. 4, pp. 1–6, 2013.
- [24] H. I. Kim, Y. K. Kim, and J. W. Chang, "A grid-based cloaking area creation scheme for continuous LBS queries in distributed systems," *Journal of Convergence*, vol. 4, no. 1, pp. 23–30, 2013.

## Research Article

# Analysis and Enhancement of IEEE 802.15.4e DSME Beacon Scheduling Model

**Kwang-il Hwang and Sung-wook Nam**

*Department of Embedded Systems Engineering, Incheon National University, Incheon 402-772, Republic of Korea*

Correspondence should be addressed to Kwang-il Hwang; [hkwangil@incheon.ac.kr](mailto:hkwangil@incheon.ac.kr)

Received 23 January 2014; Accepted 6 April 2014; Published 6 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 K.-i. Hwang and S.-w. Nam. 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 order to construct a successful Internet of things (IoT), reliable network construction and maintenance in a sensor domain should be supported. However, IEEE 802.15.4, which is the most representative wireless standard for IoT, still has problems in constructing a large-scale sensor network, such as beacon collision. To overcome some problems in IEEE 802.15.4, the 15.4e task group proposed various different modes of operation. Particularly, the IEEE 802.15.4e deterministic and synchronous multichannel extension (DSME) mode presents a novel scheduling model to solve beacon collision problems. However, the DSME model specified in the 15.4e draft does not present a concrete design model but a conceptual abstract model. Therefore, in this paper we introduce a DSME beacon scheduling model and present a concrete design model. Furthermore, validity and performance of DSME are evaluated through experiments. Based on experiment results, we analyze the problems and limitations of DSME, present solutions step by step, and finally propose an enhanced DSME beacon scheduling model. Through additional experiments, we prove the performance superiority of enhanced DSME.

## 1. Introduction

Growing concerns about machine-to-machine communications, such as sensor networks and the Internet of things (IoT), have accelerated the development of a low-power, low-rate, and low-cost wireless system. In particular, IEEE 802.15.4 [1] has become a representative standard for low-rate wireless personal area network (LR-WPAN) communications. One of the main advantages in using IEEE 802.15.4 is low-power operation, which is accomplished by a beacon-enabled mode. However, the beacon-enabled mode requires all devices in the network to be synchronized with a pan coordinator (PC) and this PC manages a superframe to maintain active durations and inactive durations within beacon intervals. This requirement also makes it difficult for a WPAN to be extended to multihop networks [2–4]. One of the critical problems is collision among beacon frames transmitted by different devices.

Therefore, beacon scheduling to address the beacon collision problem has been considered one of the significant

challenges in multihop networks comprising IEEE 802.15.4 systems. In order to address the beacon collision problem, various beacon scheduling methods have been studied, so far. Beacon scheduling can be largely categorized into the tree-based approach and the mesh-based approach. The tree-based category includes various beacon scheduling approaches [5–11] based on a tree network topology. The tree network has the advantage of low routing overhead, but it is prone to partial network isolation from link failure in an intermediate node. On the other hand, a mesh network can provide more flexible topology management, and thus some research [12–16] emphasizes the importance of beacon scheduling for mesh networks. In particular, an IEEE 802.15.4 task group (TG4e) realized the need for beacon scheduling in mesh networks, and they evolved an efficient beacon scheduling model utilizing a specific bitmap for neighboring superframe duration slot management in the IEEE 802.15.4e draft [15], more specifically a deterministic and synchronous multichannel extension (DSME) capability. However, the draft provides only a concept, so concrete algorithm details

TABLE 1: Tree-based beacon scheduling.

| Title (author)    | Characteristics                                                                                                                                                    |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Cho and An [5]    | Proposes a clustering approach based on group management using transmission power control                                                                          |
| TG4b [6]          | Proposes two approaches:<br>(i) using a dedicated beacon-only period in a superframe<br>(ii) compensating beacon offset between parent and child                   |
| Koubâa et al. [7] | Proposes time division beacon scheduling based on the neighbors' location and connection information                                                               |
| Ahn et al. [8]    | Proposes a slot allocation method coupled with the Cskip address allocation scheme introduced in ZigBee                                                            |
| Yeh et al. [9]    | Utilizes separate uplink and downlink slots                                                                                                                        |
| Yen et al. [10]   | Proposes stochastic beacon scheduling in order to reuse beacon slots in the network                                                                                |
| Chen et al. [11]  | Proposes application specific beacon scheduling for sensor networks<br>(i) controls the beacon interval adaptively according to varying the target detection level |

TABLE 2: Mesh-based beacon scheduling.

| Title (author)            | Characteristics                                                                                                             | Remarks                                                                                                                     |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Burda and Wietfeld [12]   | Proposes a random beacon slot assignment to construct autonomous and distributed mesh networks using a beacon-enabled mode. | (i) Requires a long time delay to complete an integrity check.<br>(ii) Has high collision probability among request packets |
| IEEE 802.15.5 [13]        | Mesh network based on IEEE 802.15.4<br>(i) characterizes block address assignment<br>(ii) utilizes a connection matrix      | Asynchronous energy saving in non-beacon-enabled mode might create unnecessary energy waste and delay                       |
| MeshMAC [14]              | Distributed beacon scheduling for a mesh network                                                                            | Does not address how to collect a neighboring beacon slot                                                                   |
| IEEE P802.15.4e DSME [15] | Provides a concept level for bitmap-based distributed beacon scheduling                                                     | (i) Does not address SD slot selection methods<br>(ii) Correctness of the algorithm is not yet verified via experiments     |
| DFBS [16]                 | Distributed, fast beacon scheduling for a mesh network<br>(i) bitmap-free beacon scheduling                                 | (i) Proposing a conceptual model<br>(ii) Similar to IEEE 802.15.4e DSME except for using an indicator instead of a bitmap   |

for implementation as well as validity of the algorithm through experiments have not yet been provided.

In this paper, through experiments, we evaluate the validity of the DSME beacon scheduling model specified in the IEEE 802.15.4e draft and propose a concrete design model. The experiment was conducted using ns-3 [17], a popular open source network simulator, and we found some problems in the pure DSME beacon scheduling algorithm by analyzing the experiment results. Therefore, based on the results, we revised the pure DSME beacon scheduling algorithm step by step and now propose enhanced DSME beacon scheduling, including limited permission notification coupled with a proper superframe duration slot-selection method.

The remainder of this paper is organized as follows. Section 2 discusses various beacon scheduling algorithms in multihop networks. In Section 3, we present an overview of IEEE 802.15.4e (an enhanced draft of IEEE 802.15.4). Experimental study of the DSME beacon scheduling is presented in Section 4. Section 5 introduces the enhanced DSME beacon scheduling, and performance is evaluated in Section 6. Finally, Section 7 provides concluding remarks.

## 2. Related Work

Over the past few years, several researchers have made numerous attempts to construct scalable multihop WPANs

based on a beacon-enabled mode and having the advantage of energy efficiency. Major challenges in a multihop extension are synchronization and collision avoidance. In order to address these two problems, various beacon scheduling methodologies have been proposed. The beacon scheduling algorithm can also be divided into two categories according to topology: tree-based beacon scheduling and mesh-based beacon scheduling.

Even though research on tree-based beacon scheduling as presented in Table 1 resulted in various attempts to solve the beacon collision problem for multihop-enabled WPANs, all of these methods only focus on the tree topology, and thus it is impossible to apply them to another topology, such as a mesh structure. The entire tree structure might be reconstructed if a communication link failure on the path of the tree occurs. Moreover, interference with other communications might occur because each node manages only its uplink and downlinks. These problems restrict utilization of a multihop low-power WPAN in more applications. Table 2 presents research on mesh-based beacon scheduling.

## 3. Overview of IEEE 802.15.4e DSME

As an enhanced version of IEEE 802.15.4, IEEE 802.15.4e includes new network structures and functionalities to meet a variety of application requirements in LR-WPANs. To

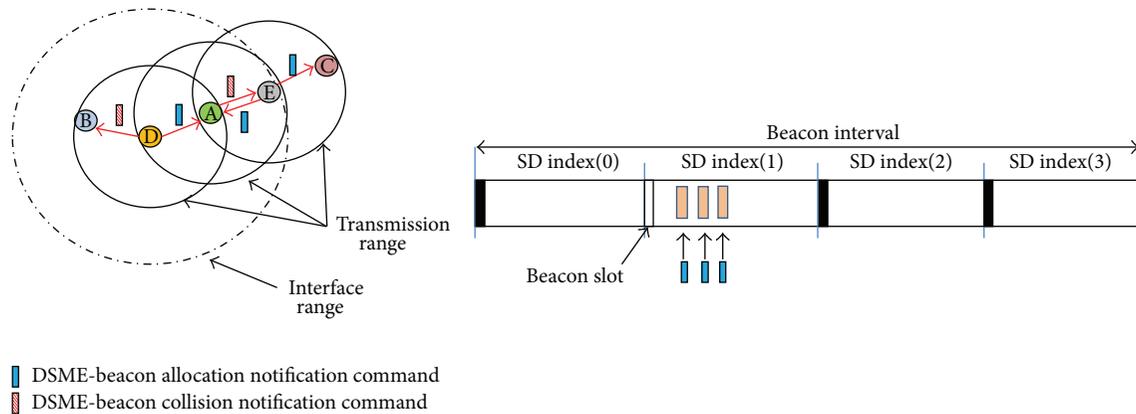


FIGURE 1: An example of collision avoidance during SD slot allocation in IEEE 802.15.4e DSME.

accommodate these requirements, IEEE 802.15.4e provides five different types of mode as follows:

- (i) TSCH: time-slotted channel hopping for high throughput requirements,
- (ii) LLDN: low latency deterministic network for high reliability and low latency,
- (iii) DSME: deterministic and synchronous multichannel extension for deterministic latency and scalability requirements,
- (iv) RFID: radio frequency identification (e.g., Blink) for item and people identification, location, and tracking,
- (v) AMCA: asynchronous multichannel adaptation for infrastructure monitoring networks.

From the above modes, DSME involves a novel beacon scheduling for mesh networks. Even though DSME supports multichannel utilization, the multichannel operation is limited only in a contention-free period (CFP) for guaranteed time slot (GTS) usage. Actual superframe operation is achieved using a single channel, so DSME focuses on avoiding beacon collisions among different WPANs by introducing a beacon scheduling method in which different pan coordinators in a complex mesh network are synchronized by conducting beacon scheduling based on a multisuperframe structure, which allows a number of superframes to coexist in a beacon interval (BI). In DSME beacon scheduling, each prospective device (more specifically, the pan coordinator) first performs a scan procedure over the available channels. Each DSME device has a superframe duration (SD) index table to manage the SD information of neighboring nodes. In addition, the SD index information of a node is represented as a bitmap, included in a macSDBitmap field of the beacon frame, which is transmitted periodically to notify neighbors about current SD index allocation information. If a prospective node receives a beacon of an active node indicating that an SD index is already allocated, the node selects a vacant slot, which is represented as "0" in the received macSDBitmap, sets the corresponding bit to "1", and broadcasts a DSME beacon-allocation notification command frame to its neighbors.

The neighboring nodes that receive the notification command first check if the bit is being used by other neighboring nodes and then they update their SD index table if the slot is available.

However, in the SD index allocation process mentioned above, a collision might occur when more than two devices make an attempt to occupy the same slot. Figure 1 illustrates this beacon collision situation. When nodes D and E, which are neighbors of node A but cannot communicate with each other, receive a beacon from node A, both can select the same slot out of vacant slots in the SDBitmap (the hidden-node problem). In that case, the two nodes have the same SD index, so the beacon transmission slot overlaps. That is, since the beacons of the two nodes collide, node A cannot hear either beacon transmitted from the two nodes. To address this problem, DSME uses an additional frame, a DSME beacon-collision notification command. If the two nodes want to use the same SD index by sending an allocation notification message, node A allows the node arriving first to allocate the SD slot, and if another node then requests the already occupied slot, node A makes the new requester select another slot by sending the collision notification command. Eventually, this procedure can avoid overlapped allocation of SD indexes among neighboring nodes. This method provides a simple but powerful beacon scheduling, which is not solved in IEEE 802.15.4. In particular, it is possible for a superframe duration of two-hop neighboring nodes as well as neighboring nodes to be scheduled in a distributed manner.

#### 4. Experiment with IEEE 802.15.4e DSME Beacon Scheduling

IEEE 802.15.4e DSME beacon scheduling, as described in the previous section, can provide efficient scheduling among neighboring WPANs. However, the standard introduces just an abstract concept without any concrete outline and implementation details. Therefore, in this section we first present a guideline for implementation of DSME beacon scheduling and then evaluate the performance and validity of the algorithm.

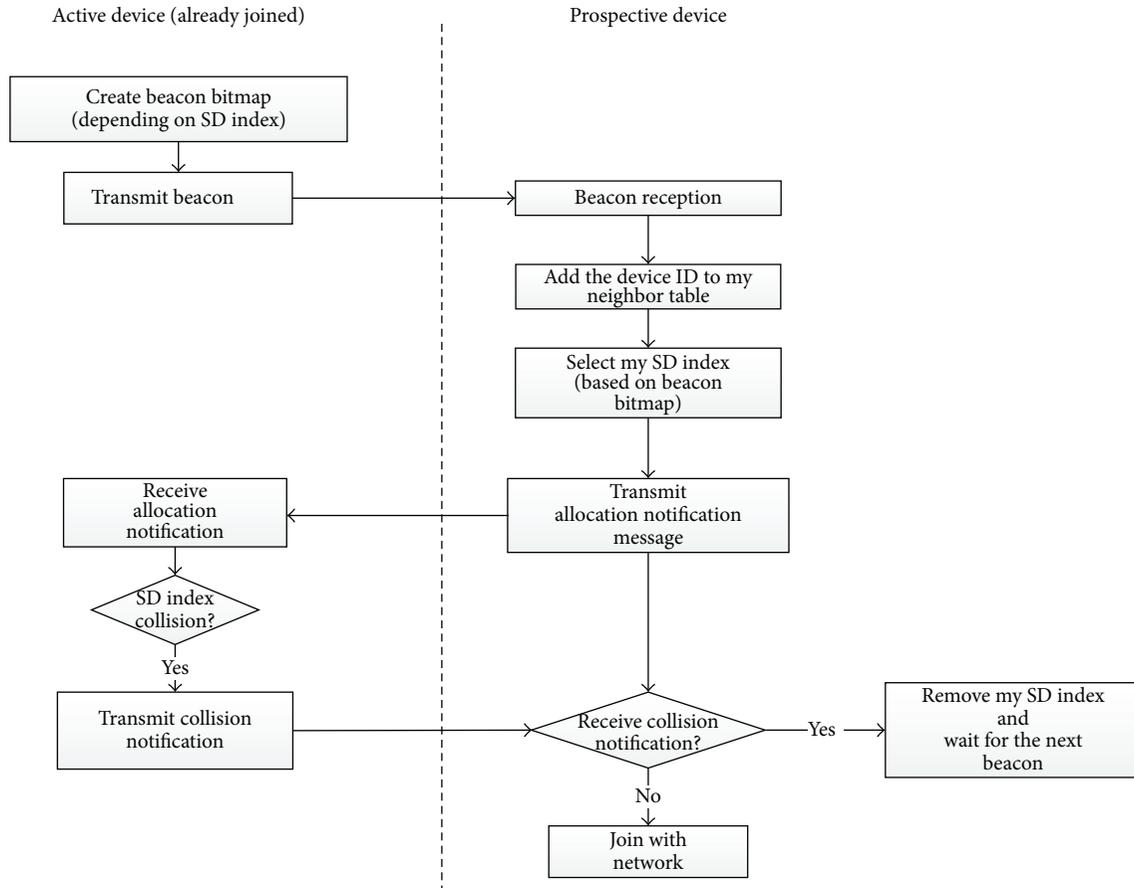


FIGURE 2: Flowchart of the IEEE 802.15.4e DSME beacon scheduling algorithm.

**4.1. Experiment Environment.** We implemented beacon scheduling for IEEE 802.15.4 DSME based on the LR-WPAN module of ns-3 [17]. For a channel model, a single model spectrum channel was used and a communication channel was constructed based on a long distance propagation loss error model. Specific primitives and frames for DSME are involved in the LR-WPAN media access control (MAC) layer and additional PAN information base (PIB) attributes and constants are defined in the MAC header. In addition, instead of a superframe structure used in the original IEEE 802.15.4, a multisuperframe structure (a specific feature of DSME) is used.

#### 4.2. Pure DSME

**4.2.1. DSME Beacon Scheduling Review.** This subsection presents the DSME beacon scheduling algorithm in terms of implementation. Figure 2 shows a concrete flow chart of a DSME beacon scheduling algorithm. The algorithm is divided into two stages: the active node stage and the prospective node stage. First, an active node generates a beacon bitmap based on its own SD index and the SD index information of its neighboring nodes. This SDBitmap information is contained in a beacon frame broadcasted periodically. All nodes update their neighboring SD index table whenever

they receive a beacon frame containing an SDBitmap field. In particular, if a prospective node receives a beacon frame, it adds the sender ID to the SD index table entry and updates SD index information based on the SDBitmap contained in the received beacon. Then, a prospective node selects an available (vacant) slot and broadcasts a beacon allocation notification message. All the nodes that receive the allocation notification message check whether the notified slot is already used. If the slot is available, the SD index table is updated by adding the designated slot. However, if the slot is already used, the node that receives the notification message sends a collision notification message to the originator to prevent overlapped SD slot usage. As soon as the prospective node, which sent notification of its new SD index, receives a collision notification message, it clears the SD index and then waits until the next beacon reception. After broadcasting the allocation notification message, if a collision notification message is not received by the end of the contention access period (CAP) of the present superframe duration, the prospective node becomes an active node with the allocated SD index, and it can send its periodic beacon in the allocated SD index slot.

**4.2.2. IEEE 802.15.4e DSME SD Slot Selection.** If a prospective node receives a beacon, a superframe slot of the node can be allocated by selecting a vacant slot. The allocated slots are

```

(Variables)
A[]: BITMAP ARRAY
sA: Size of bitmap data type
(eg., unsigned char A[BITMAP_ARRAY_SIZE],
in that case we have sA = 8)
index: indicator of array index
position: bit position in a A[x],
where $x = 0, 1, 2, \dots, \text{BITMAP_ARRAY_SIZE}$

LAB (A)
(1) index $\leftarrow 0$
(2) for i $\leftarrow 0$ to BITMAP_ARRAY_SIZE
(3) for j $\leftarrow 0$ to sA - 1
(4) if (A[i] & (0 × 01 << j)) == 0
(5) SD_index $\leftarrow j + (\text{sA} * i)$
(6) return SD_index
(7) j $\leftarrow j + 1$
(8) i $\leftarrow i + 1$
(9) return NO_AVAILABLE_SD

MAB (A)
(1) index $\leftarrow (\text{BITMAP_ARRAY_SIZE} - 1) / \text{sA}$
(2) position $\leftarrow (\text{BITMAP_ARRAY_SIZE} - 1) \bmod \text{sA}$
(3) if (A[index] & (0 × 01 << position)) == 1
(4) return NO_AVAILABLE_SD
(5) for i \leftarrow index to 0
(6) for j $\leftarrow 0$ to sA - 1
(7) if (A[i] & ((0 × 01 << sA - 1) >> j)) == 0
(8) SD_index $\leftarrow \text{sA} - j + (j * \text{sA})$
(9) return SD_index
(10) j $\leftarrow j + 1$
(11) i $\leftarrow i - 1$
(12) return NO_AVAILABLE_SD

RAND_SD (A)
(1) do
(2) r_sd \leftarrow random(1.MAX_SD)
(3) index \leftarrow r_sd / sA
(4) position \leftarrow r_sd mod sA
(5) while (A[index] & (0 × 01 << position)) == 1
(6) SD_index \leftarrow position + index * sA
(7) return SD_index

```

ALGORITHM 1: Possible SD slot selection algorithms.

represented as “1” in the SDBitmap so that the node chooses a slot from the “0” bits. As a matter of fact, since the allocation method is distributive and the SDBitmap only represents the SD index of the neighboring sender, slot allocation distribution in the received SDBitmap at any time varies according to network topology. Therefore, to make a rule for selecting an SD index from vacant slots in a bitmap, we consider three different selection methods: least available bit (LAB), most available bit (MAB), and random. However, we also presume that the beacon scheduling performance might vary according to the selection method used.

Algorithm 1 shows each of the three possible SD slot selection algorithms: LAB, MAB, and random. LAB searches the first “0” bit for the received bitmap from the least significant bit. The first “0” bit finally becomes its own beacon slot number. On the other hand, MAB searches the first “1”

bit for the received bitmap from the most significant bit. The “0” bit followed by the first “1” becomes its own beacon slot number. Random method randomly chooses one number and then the selected number is used if the corresponding bit is clear in the bitmap. Otherwise, the random selection process is repeated until the selected bit is clear.

The LAB selection method is to choose the lowest slot number out of the vacant slots. This method might increase the reuse ratio of the superframe duration among the nodes that are separated by more than two hops. However, there is the possibility that collisions will occur during the SD index selection process.

The MAB selection method is to choose the vacant bit that immediately follows the largest value of the allocated slot numbers. This method may provide a lower reuse ratio of the SD index than LAB, so the possibility of collision might be reduced. In addition, with more hops, there is a greater possibility of allocating an SD index in order.

The random method chooses a vacant slot at random. Random SD index selection might increase unnecessary network traffic to avoid collisions because of the possibility that different nodes select the same slot.

**4.2.3. Experiment Results.** In this subsection, through the experiments we verify the validity of DSME and evaluate its performance with respect to different SD index slot selection methods. For the experiments, we considered two representative topologies: sparse and dense models, consisting of  $3 \times 3$  nodes as shown in Figure 3.

To evaluate the performance of DSME beacon scheduling, we first observed the successful SD slot allocation ratio with respect to different SD slot selection methods (LAB, MAB, and random) in sparse and dense topology models, as shown in Figure 4. The result shows that the MAB selection method is superior to LAB and random. In particular, the LAB selection method shows the worst performance. That is because LAB caused a number of collisions among the nodes that chose the same SD slot, as we expected. However, the result also reveals that even MAB, which shows the best performance among them, shows an allocation failure ratio of more than 20 percent in the dense topology. This results from a beacon collision problem that is not yet completely resolved.

Through experiments, we also found limitations of pure DSME in some specific situations. The problem is mainly caused by collisions of command messages transmitted from different nodes at the same time. In particular, the more complex the topology and the more the devices, the higher the collision possibility. Figure 5 illustrates an example of collision among allocation notification messages during an SD allocation phase. As shown in Figure 5(a), nodes 1, 2, 4, and 5 have completed SD index allocation and, as a result, slots 3, 4, and 5 are allocated for nodes 2, 4, and 5, respectively. After finishing the superframe duration of node 1, node 5 (which has slot 2) transmits its beacon, as shown in Figure 5(b). Already activated nodes 1, 2, and 4 just update their neighbor SD index table. On the other hand, the remaining nodes where the SD index is not yet allocated select an available SD index based on the received

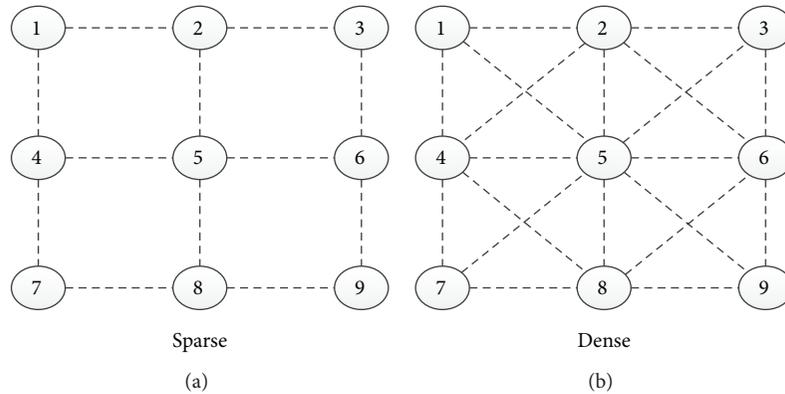


FIGURE 3: Experiment topologies: sparse and dense models.

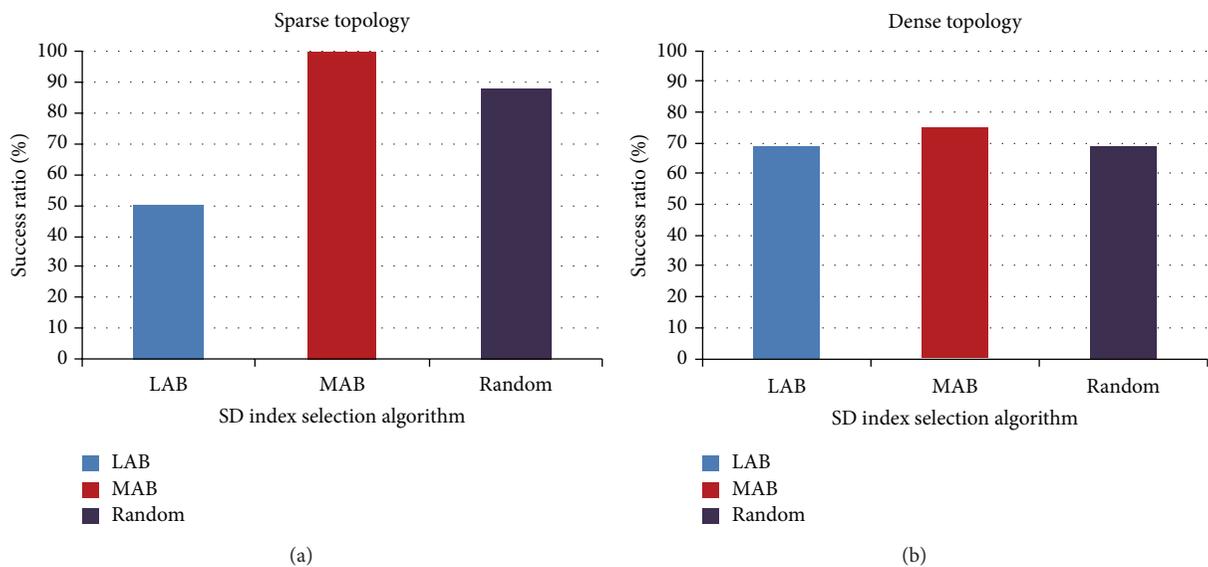


FIGURE 4: Experiment results of pure DSME beacon scheduling.

SDBitmap. Since the nodes receive the same SDBitmap and use the same SD selection algorithm, they choose the same vacant slot and make an attempt to transmit their own allocation notification message using carrier sense multiple access/collision avoidance (CSMA/CA). Even though they transmit their own messages using CSMA/CA, collisions still occur, as shown in Figure 5(c), and thus the nodes that are not acknowledged by node 5 retry sending the message after waiting for random back-off delay. However, since the random back-off duration is lengthened more whenever collision occurs, eventually the present superframe duration being maintained by node 5 is not completed and some nodes remain unallocated. That is, when the topology is complex and the number of nodes increases, the collision possibility for command messages increases. This results in SD index allocation failure in some nodes. Furthermore, if acknowledgement of each command frame is disabled, after transmitting its notification message, each node is convinced that its selected SD slot is available and transmits its beacon

on the allocated SD slot. In that case, the beacon frames of the nodes might collide with each other, as shown in Figure 5(d).

The major reason for these collisions is the hidden node problem, in which each node cannot identify the presence of other nodes. Therefore, even though every node performs CSMA before a transmission, allocation notification messages of others might not be detected. Furthermore, the back-off effect with respect to the same beacon frame results in an increased collision possibility among messages. To observe the effect of hidden node problems in a realistic environment, we conducted an experiment on successful data ratios with respect to varying the number of hidden nodes, as shown in Figure 6. The result shows that a 100% success ratio is not guaranteed, even among two hidden nodes, and performance deteriorates drastically as the number of hidden nodes increases.

*4.3. Distributed Permission Notification.* The experiment results revealed that the DSME beacon scheduling algorithm

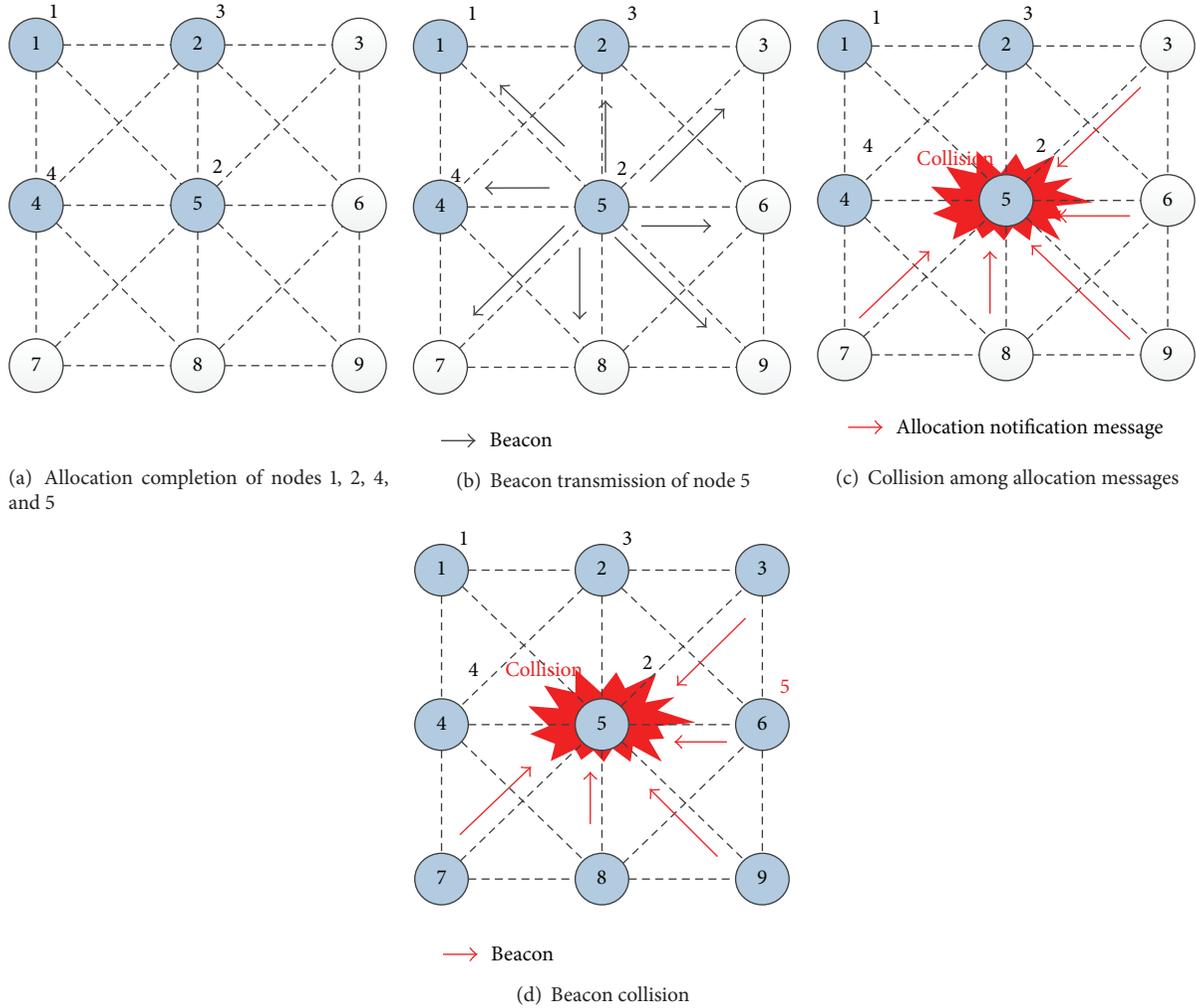


FIGURE 5: An example of collision among allocation notification messages.

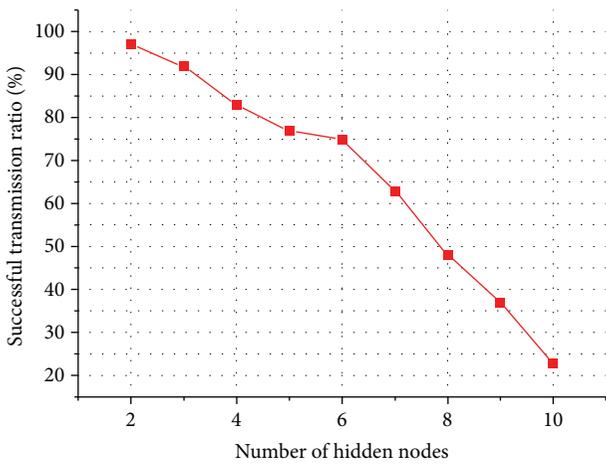


FIGURE 6: Number of hidden nodes versus packet success ratio.

has some critical problems. Therefore, it is necessary to rework the pure DSME beacon scheduling algorithm by

resolving collisions. One of the significant problems in pure DSME is collisions between command frames, such as allocation notification and collision notification, and another is beacon collision that is caused by using overlapped SD slots. This is because a node allocates its SD index slot for itself based on the received SDBitmap information.

So, we first employ a distributed permission notification to enhance collision notification specified in pure DSME. While pure DSME uses a negative allocation by sending a collision notification only when the newly allocated SD slot overlaps with another neighbor's, the distributed permission method uses positive allocation by allowing only the node that receives a permission notification message after sending an allocation notification message to complete the SD index allocation. A prospective node that sent an allocation notification message waits until permission notification is received, and the neighboring active node that receives allocation notification from the prospective node checks whether the requested SD index is available or not, and if the slot is available, it broadcasts a permission notification. The neighbors of the active node that broadcast permission

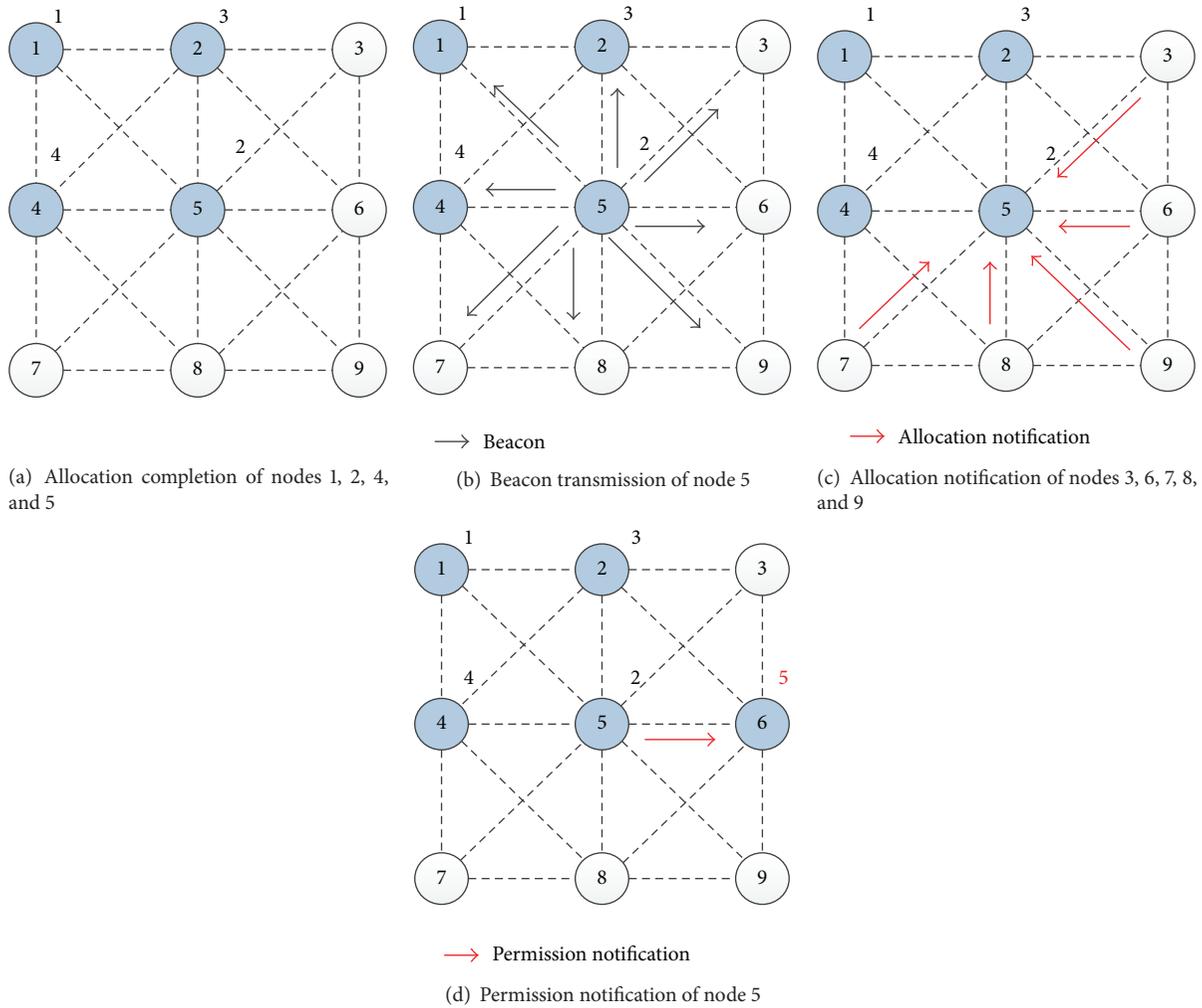


FIGURE 7: An example of distributed permission notification method.

notifications also come to know the information of the SD index of a new node separated by two hops by hearing this permission notification. This might result in the reduction of collision. Furthermore, it is even expected to cope well with a complex topology. Figure 7 shows an example of a distributed permission method. As in the previous example, the nodes that receive the beacon from node 5 send their allocation notification message to node 5. However, unlike pure DSME, where there have been collisions among the allocation notification messages, the permission method can avoid the collisions by only permitting the allocation of node 6, which is selected by node 5.

This enhancement is expected to significantly improve performance of DSME. However, contrary to our expectations, the experiment results were worse than pure DSME. As shown in Figure 8, the successful beacon scheduling ratio shows lower values in both sparse and dense models.

Figure 9 illustrates the main reason for the performance degradation. The distributed permission method can avoid collisions by learning the SD index information of two-hop nodes through permission notification, but all the nodes

that receive an allocation notification message have the right to send a permission notification, and thus, as shown in Figure 9, the nodes that do not have to be allocated also complete SD allocation. Therefore, they allocate the same SD slot, and eventually beacon collision occurs. The collisions also occur regardless of the SD index selection method used.

In addition to that reason, there is the possibility that the permission notification messages of active nodes might collide with the allocation notifications of the prospective nodes, so that the neighboring active nodes often miss the SD index information of new neighboring nodes.

## 5. Enhanced DSME Beacon Scheduling

5.1. DSME Experiment Results Review. IEEE 802.15.4e DSME beacon scheduling presents a method that can minimize beacon collisions efficiently using a multiple superframe structure in mesh-based multihop networks. However, experiment results demonstrated that the DSME beacon scheduling model is still in its conceptual stage and needs to be enhanced in order to apply it to various topology models

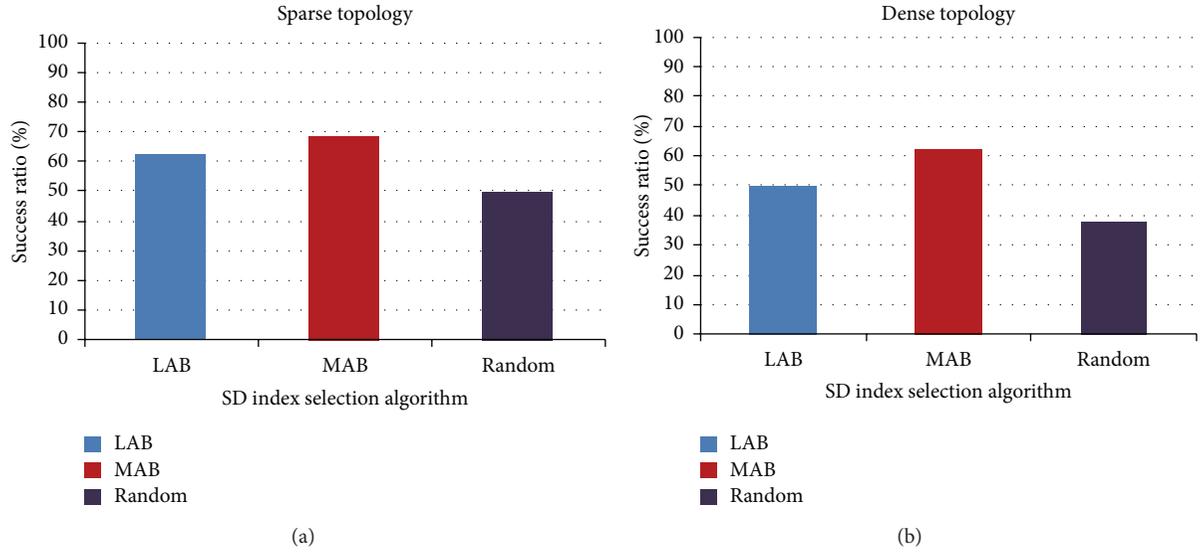


FIGURE 8: Experiment results of the distributed permission method.

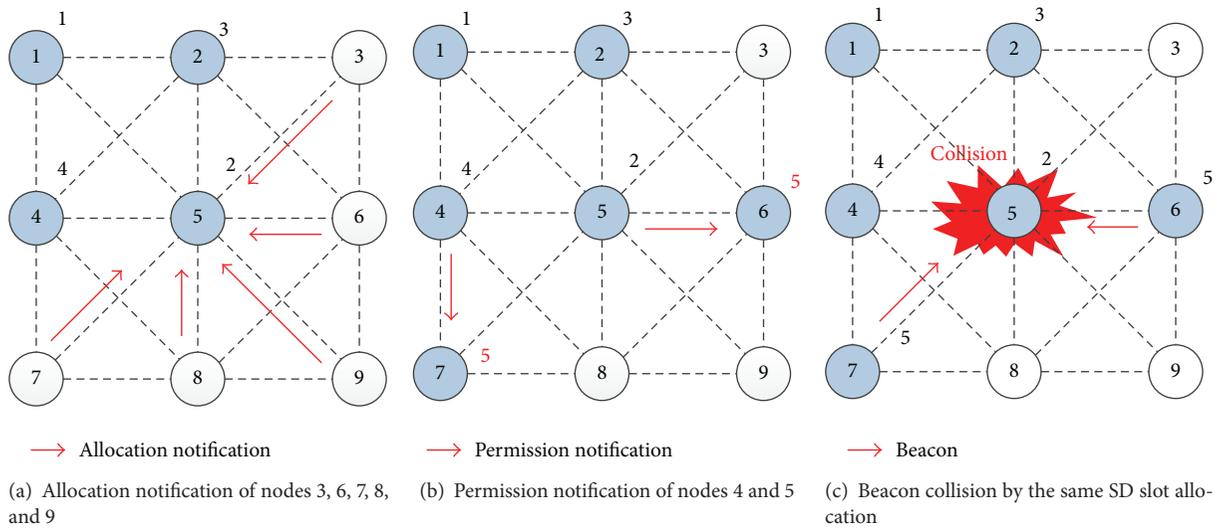


FIGURE 9: Main reason for performance degradation: distributed permission method.

and environments. In the previous section, we analyzed the DSME beacon scheduling based on experiment results and proposed the first revision in which we tried to resolve command frame collision problems caused by absence of information about nodes separated by two hops. Even though our first revision can solve the problems caused in pure DSME, overall performance was not improved because it has another collision problem. Therefore, in this section, we present a new revision of DSME beacon scheduling to cope well with several problems that are not solved so far and we verify the algorithm's correctness and performance.

**5.2. Limited Permission Method.** The distinguished enhancements of enhanced DSME (E-DSME) are the use of limited permissions and a new superframe structure suitable for

distributed beacon scheduling. First, a permission right is only limited to the originator of the latest beacon, as shown in Figure 10. Restricting the node that can send permission to the sender of the beacon can avoid abuse of SD allocation caused by unnecessary permission notifications from neighboring active nodes. In addition to this, the superframe is restructured, as shown in Figure 11. A superframe duration is composed of a number of SD allocation durations (SAD), which also consist of an allocation contention period (ACP) and a permission notification period (PNP). ACP is a period in which prospective nodes that receive a beacon of a parent PC assign their own candidate SD index and transmit an allocation notification to the parent PC through contention. A permission notification of a parent PC is allowed only during the PNP. This is to separate transmission timings

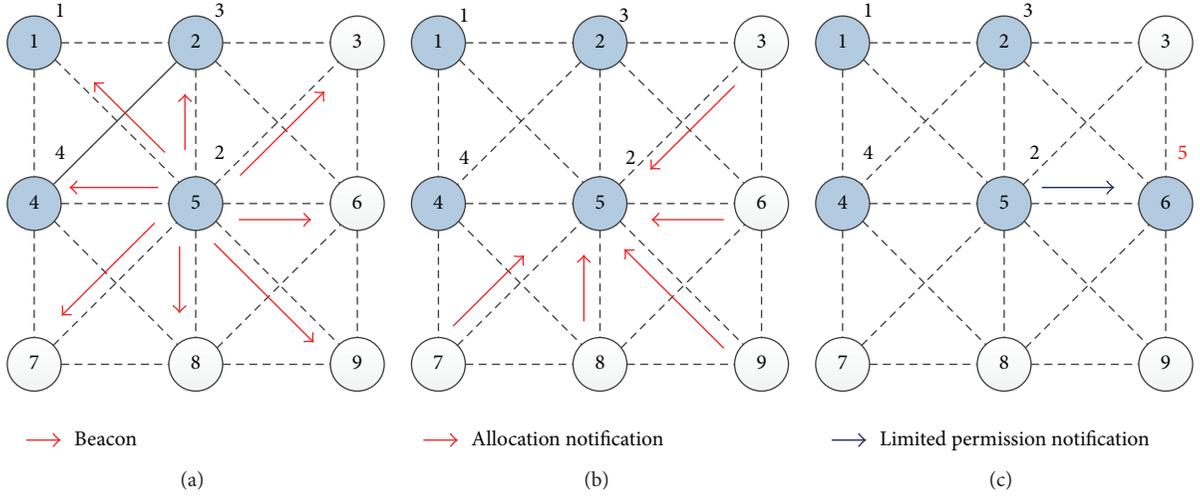


FIGURE 10: Limited permission notification.

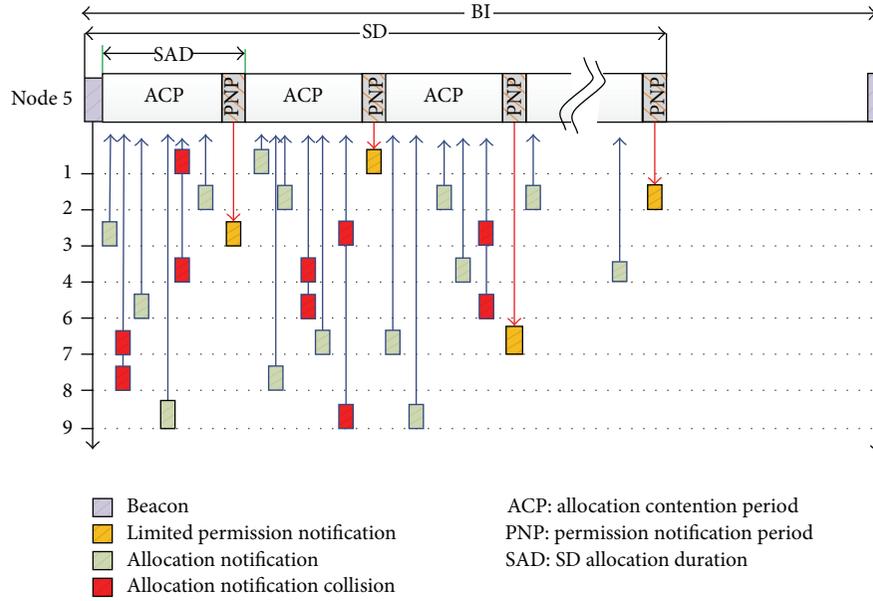


FIGURE 11: E-DSME superframe structure.

between allocation notification and permission notification messages. The repetitive architecture of SAD within a superframe duration enables a node that did not have permission during the first SAD to update its candidate SD index by itself and retry allocation notification with the updated SD index. Therefore, it is possible for more prospective nodes to join the network within a superframe duration.

5.3. *E-DSME Superframe Structure.* As mentioned in the previous subsection, a superframe duration is made up of a number of SADs. A SAD is also composed of ACPs and PNPs:

$$SAD = T_{ACP} + T_{PNP}. \tag{1}$$

Here, each duration of  $T_{ACP}$  and  $T_{PNP}$  is as follows:

$$T_{ACP} = \left[ \left\{ \left( \sum_{i=0}^{\text{macMaxBE}} 2^{\text{maxMinBE}+i} \right) \times \text{aUnitBackoffPeriod} \right\} + \text{aBaseSlotDuration} \right] \times (\text{Symbol rate})^{-1}, \tag{2}$$

$$T_{PNP} = \frac{\text{aBaseSlotDuration}}{\text{Symbol rate}}.$$

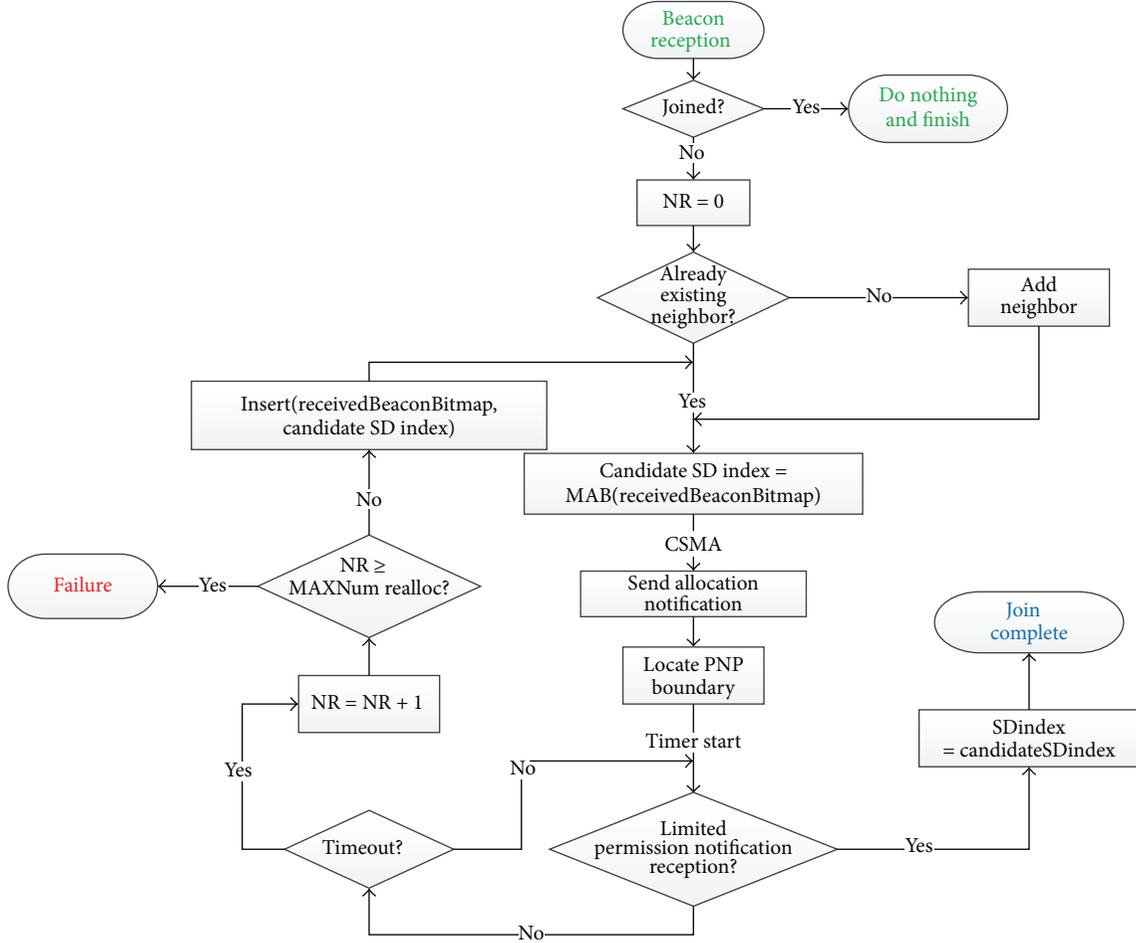


FIGURE 12: E-DSME beacon scheduling.

In addition, the maximum number of SADs in a superframe is obtained as follows:

$$\text{MaxNumSAD} = \min \left\{ \frac{\text{SD}}{\text{SAD}}, \text{macMaxBeaconBitmapSize} - \text{CandidateSDindex} \right\}, \quad (3)$$

where  $\text{SD} = \text{aBaseSuperframeDuration} \times 2^{\text{SO}}$ ,  $0 \leq \text{SO} \leq 14$  and  $\text{aBaseSuperframeDuration} = \text{aBaseSlotDuration} + \text{aNumSuperframeSlots}$ .

**5.4. E-DSME Beacon Scheduling.** Figure 12 shows a beacon scheduling flowchart for E-DSME. Upon the reception of a beacon frame from a neighbor, a prospective device first checks whether it has already joined or not. In addition, the originator of the received beacon is not registered in neighbor table, the information is updated, and then the prospective node selects a candidate SD index from the received bitmap using MAB slot selection method. To avoid collisions, the node waits for the upcoming PNP after transmitting allocation notification request. If, during the PNP, there is no

permission, the node retries allocation notification request at the next ACP. If a permission notification is received from the originator, the node registers current candidate SD index slot for its own SD index slot and then completes the join procedure. The outstanding feature of E-DSME beacon scheduling is that, according to success or failure, a prospective node can update its own candidate SD index by itself and perform an allocation notification procedure repeatedly.

**5.5. Algorithm Verification.** Testing enhanced DSME was conducted in the same environment as previous experiments. As shown in Figure 13, enhanced DSME shows a 100% allocation success ratio when the MAB SD index selection method is applied in both the sparse and the dense models. Compared to the previous method (pure DSME and distributed permission notification), performance improvement with enhanced DSME beacon scheduling might result from utilizing a limited permission notification and repetitive SAD structure. Furthermore, the result demonstrates that the MAB SD index selection algorithm is the most suitable for DSME beacon scheduling compared to the other SD index selection methods, LAB and random.

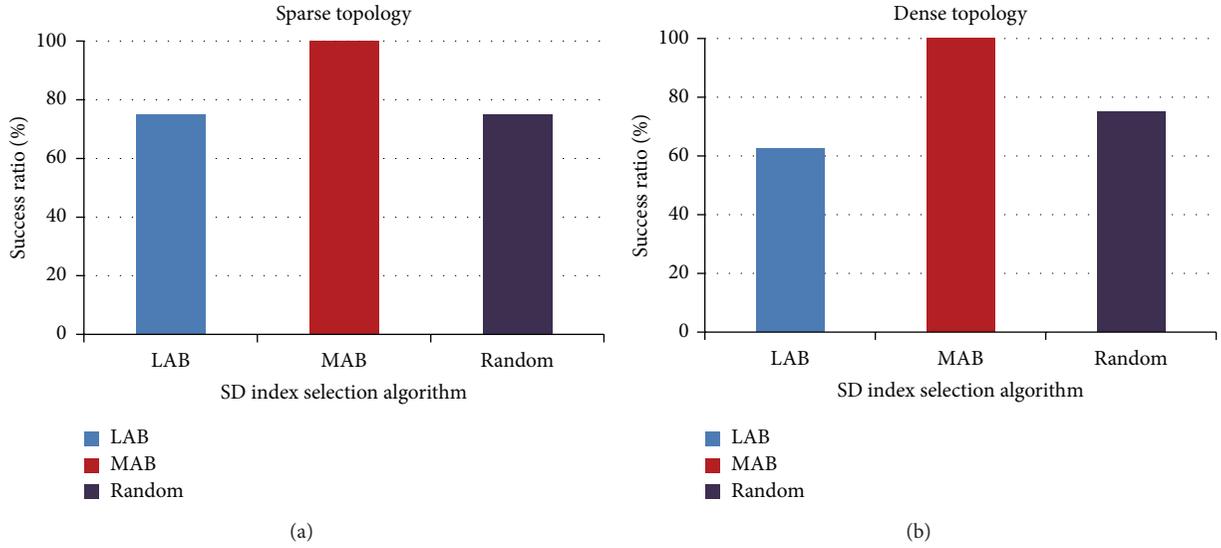


FIGURE 13: Experiment results of E-DSME beacon scheduling.

TABLE 3: Key parameters used in the experiments.

| Parameter                               | Value                                    |
|-----------------------------------------|------------------------------------------|
| <i>Bitrate</i>                          | 250 kb/s                                 |
| <i>Symbol rate</i>                      | 62.5ksymbol/s                            |
| <i>aMaxPHYPacketSize</i>                | 127                                      |
| <i>phyCurrentChannel</i>                | 0                                        |
| <i>phyCCAMode</i>                       | 1                                        |
| <i>aBaseSlotDuration</i>                | 60 symbols                               |
| <i>aBaseSuperframeDuration</i>          | $aBaseSlotDuration * aNumSuperframeSlot$ |
| <i>aMaxPermissionOnlyPeriodDuration</i> | $aBaseSuperframeDuration/2$              |
| <i>aNumSuperframeSlots</i>              | 16                                       |
| <i>aUnitBackoffPeriod</i>               | 20 symbols                               |
| <i>macBeaconOrder</i>                   | 0–15 (14)                                |
| <i>macMaxBE</i>                         | 3–8 (5)                                  |
| <i>macMaxCSMABackoffs</i>               | 0–5 (4)                                  |
| <i>macMaxFrameRetries</i>               | 0–7 (3)                                  |
| <i>macMinBE</i>                         | $0 - macMaxBE$ (3)                       |
| <i>macSuperframeOrder</i>               | 0–15 (5~7)                               |
| <i>macDSMEenabled</i>                   | (TRUE)                                   |
| <i>macMultisuperframeorder</i>          | 0–15                                     |
| <i>macBeaconSlotLength</i>              | 128                                      |

## 6. Performance Evaluation of the Enhanced DSME

**6.1. Experiment Environments.** In the previous section, we revised the pure DSME beacon scheduling step by step by analyzing experiment results. The final revision, enhanced DSME, showed a satisfactory performance in both the sparse and the dense models. However, the two topology models used in the previous experiments are so specific that we need to verify algorithm correctness and evaluate various performances of enhanced DSME via additional experiments

in which more general environments are applied. For the experiments, the number of nodes randomly deployed was also varied between 10 and 40, as shown in Figure 14. Table 3 shows the key parameters used in the experiments.

**6.2. Successful Association Ratio.** Figure 15 shows the results for successful allocation ratio with respect to varying the number of devices between 10 and 40. For comparative evaluation, we conducted enhanced DSME beacon scheduling by applying the MAB and LAB SD index selections, respectively. The result shows that, with LAB, as the number of devices

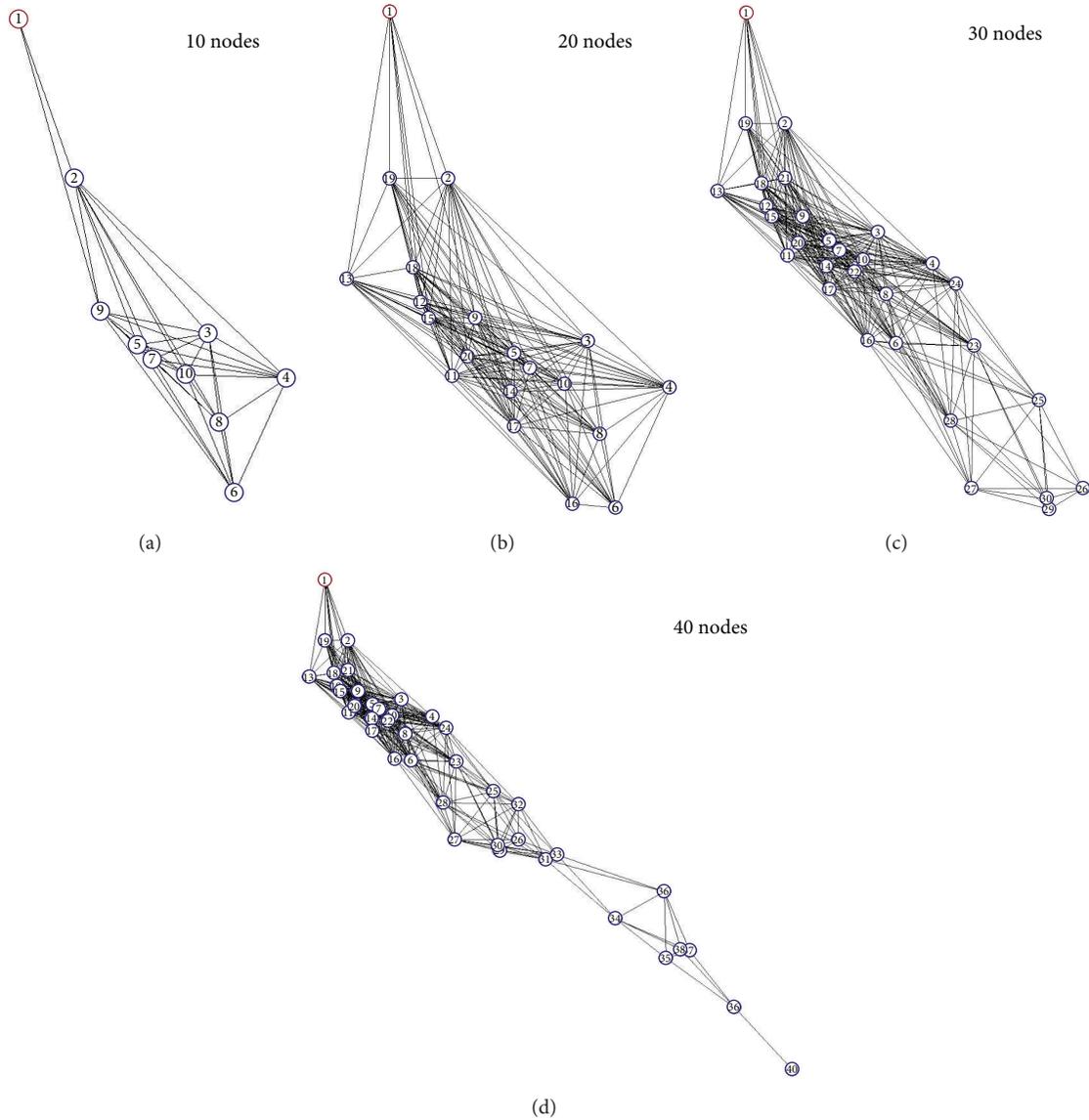


FIGURE 14: Experiment topologies.

increases, success ratio degrades. On the other hand, the result demonstrates that MAB maintains a high success ratio regardless of the number of devices. The performance degradation with LAB is because of collisions caused by the reuse of SD index slots as the topology becomes complicated. However, since MAB always selects a candidate SD index that follows the largest value in the already occupied SD index, collisions found in LAB are avoided. Therefore, note that the enhanced DSME beacon scheduling utilizing limited permissions, coupled with the MAB SD index selection scheme, shows an optimal combination.

**6.3. Allocation Completion Time.** We also observed the allocation completion time of E-DSME beacon scheduling. Allocation completion time represents the total time required to complete SD index allocation for all nodes in the network.

We observed the allocation completion time by varying the number of nodes in a different superframe order (SO): 5, 6, and 7. As shown in Figure 16, the result shows that the completion time is longer as SO size becomes larger. This is because a larger SO can generate more SD slots within a beacon duration. Since the duration of a superframe is represented by  $aBaseSuperframeDuration \times 2^{SO}$ ,  $0 \leq SO \leq 14$ , a short superframe duration can accommodate only a small number of allocation requests, and eventually the nodes not allocated in that round have to wait until the next neighbor's beacon is received.

The number of multisuperframe slots is equal to the beacon bitmap length, and the length of a bitmap is represented by  $SDBitmaplength = 2^{(BO - SO)}$ . So, enhanced DSME beacon scheduling can apply the result after calculating the expected number of slots in advance of network formation

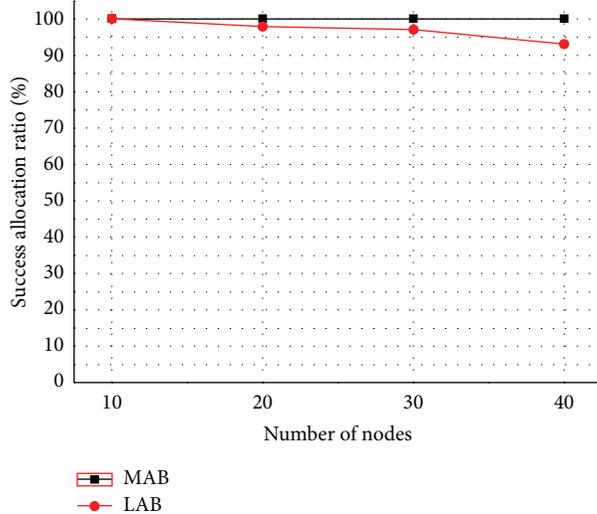


FIGURE 15: Successful allocation ratio.

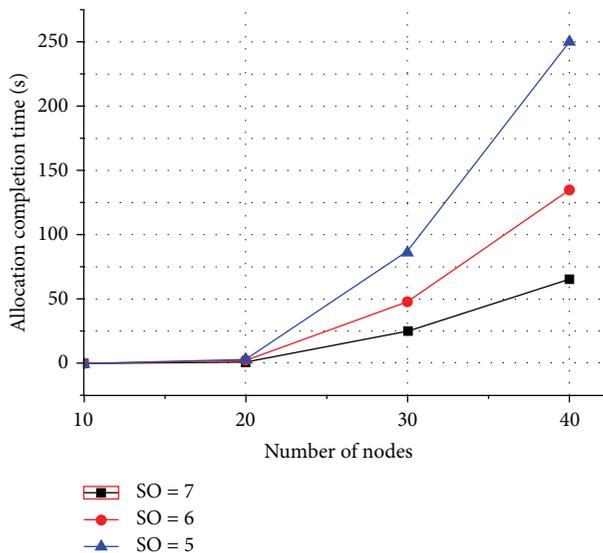


FIGURE 16: Allocation completion time.

according to the network size. Moreover, even though the network size is extended during run time, it is possible to dynamically change the expected number of slots in the process of network formation since each beacon contains BO and SO information.

## 7. Conclusion

In this paper, we introduced IEEE 802.15.4e DSME beacon scheduling, evaluated its validity and performance, and proposed a concrete design model. Through experiments, we found some problems in the pure DSME beacon scheduling algorithm by analyzing results. Therefore, based on the results, we revised the pure DSME beacon scheduling algorithm step by step and proposed an enhanced DSME beacon

scheduling including new features: limited permission notification and a repetitive SAD architecture.

The proposed E-DSME model is expected to contribute to design and modeling of beacon scheduling for large-scale sensor networks or IoT sensor domains.

## Conflict of Interests

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

## Acknowledgments

This paper is extended and improved from a paper accepted at the KCIC-2013 conference. This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A2041271).

## References

- [1] "Part 15.4: wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for low-rate wireless personal area networks (WPANs)," IEEE Standard for Information Technology IEEE Std 802.15.4, 2011.
- [2] J. Liu and S. H. Chung, "An efficient load balancing scheme for multi-gateways in wireless mesh networks," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 365–378, 2013.
- [3] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-centric Computing and Information Sciences*, vol. 3, no. 13, pp. 1–17, 2013.
- [4] M. Yoon, Y. K. Kim, and J. W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Future Technology Research Association International*, vol. 4, no. 1, pp. 15–22, 2013.
- [5] J. J. Cho and S. S. An, "An adaptive beacon scheduling mechanism using power control in cluster-tree WPANs," *Wireless Personal Communications*, vol. 50, no. 2, pp. 143–160, 2009.
- [6] M. Lee, J. Zheng, Y. Liu et al., "Combined beacon scheduling proposal to IEEE 802.15.4b," IEEE 802.15-04-0536-00-004b, September 2004.
- [7] A. Koubâa, A. Cunha, and M. Alves, "A time division beacon scheduling mechanism for IEEE 802.15.4/zigbee cluster-tree wireless sensor networks," in *Proceedings of the 19th Euromicro Conference on Real-Time Systems (ECRTS '07)*, pp. 125–135, July 2007.
- [8] S. Ahn, J. Cho, and S. An, "Slotted beacon scheduling using ZigBee Cskip mechanism," in *Proceedings of the 2nd International Conference on Sensor Technologies and Applications (SENSORCOMM '08)*, pp. 103–108, August 2008.
- [9] L. W. Yeh, M. S. Pan, and Y. C. Tseng, "Two-way beacon scheduling in ZigBee tree-based wireless sensor networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC '08)*, pp. 130–137, June 2008.
- [10] L. H. Yen, Y. W. Law, and M. Palaniswami, "Risk-aware beacon scheduling for tree-based ZigBee/IEEE 802.15.4 wireless networks," in *Proceedings of the 4th Annual International Conference on Wireless Internet (WICON '08)*, November 2008.

- [11] S. Chen, L. Almeida, and Z. Wang, "Analysis and experiments for dual-rate beacon scheduling in ZigBee/IEEE 802.15.4," in *Proceedings of the 2011 IEEE Conference on Computer Communications Workshops (INFOCOM '11)*, pp. 756–761, April 2011.
- [12] R. Burda and C. Wietfeld, "A distributed and autonomous beacon scheduling algorithm for IEEE 802.15.4/ZigBee networks," in *Proceedings of the 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS '07)*, October 2007.
- [13] M. J. Lee, R. Zhang, J. Zheng et al., "IEEE 802.15.5 WPAN mesh standard-low rate part: meshing the wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 973–983, 2010.
- [14] P. Muthukumaran, R. de Paz, R. Spinar et al., "MeshMAC: enabling Mesh Networking over IEEE 802.15.4 through distributed beacon scheduling," in *Proceedings of the the 1st International Conference on Ad Hoc Networks (AdHocNets '09)*, Niagara Falls, Canada, September 2009.
- [15] "Amendment 5: amendment to the MAC sub-layer for low-rate Wireless Personal Area Networks (WPANs) amendment 5," IEEE Standard for Information Technology IEEE P802.15.4e draft, 2011.
- [16] W. Y. Lee, K. I. Hwang, Y. A. Jeon, and S. Choi, "Distributed fast beacon scheduling for mesh networks," in *Proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS '11)*, pp. 727–732, October 2011.
- [17] ns-3, <http://www.nsnam.org/>.

## Research Article

# Optimization of High-Speed Train Control Strategy for Traction Energy Saving Using an Improved Genetic Algorithm

Ruidan Su,<sup>1</sup> Qianrong Gu,<sup>2</sup> and Tao Wen<sup>1</sup>

<sup>1</sup> College of Information Science and Engineering, Northeastern University, Shenyang 110004, China

<sup>2</sup> Service Science Research Center, Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201203, China

Correspondence should be addressed to Ruidan Su; [suruidan@hotmail.com](mailto:suruidan@hotmail.com)

Received 27 March 2014; Accepted 9 April 2014; Published 4 May 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Ruidan Su 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 parallel multipopulation genetic algorithm (PMPGA) is proposed to optimize the train control strategy, which reduces the energy consumption at a specified running time. The paper considered not only energy consumption, but also running time, security, and riding comfort. Also an actual railway line (Beijing-Shanghai High-Speed Railway) parameter including the stop, tunnel, and curve was applied for simulation. Train traction property and braking property was explored detailed to ensure the accuracy of running. The PMPGA was also compared with the standard genetic algorithm (SGA); the influence of the fitness function representation on the search results was also explored. By running a series of simulations, energy savings were found, both qualitatively and quantitatively, which were affected by applying curving and coasting running status. The paper compared the PMPGA with the multiobjective fuzzy optimization algorithm and differential evolution based algorithm and showed that PMPGA has achieved better result. The method can be widely applied to related high-speed train.

## 1. Introduction

Since October 1964 the world's first high-speed railway, Japan Tokaido Shinkansen, was born; high-speed railways started the rapid development. Today, most European countries, Russia, Japan, and China have constructed their complex high-speed railways networks. Although the railway was considered the most efficient way of travel, compared to aircraft and auto vehicle, it still consumes large amount of energy [1] in everyday running. Researches showed that it still has large possibility to make the train run more efficiently [2–4]. The reduction of energy consumption is also seen as one of the key objectives for the development of sustainable mobility by use of high-speed train. Research will lead to a decrease of huge energy consumption in everyday running of high-speed trains. Many scholars have been engaged in it.

Yang et al. [5] from Tongji University proposed a new energy conservation track profile based on trigonometric function method in urban mass transit. Simulation results showed that it was effective in comparison with actual track profile. Bocharnikov et al. [6] applied a method for saving

energy consumption during a single-train journey by trading off reductions in energy against increases in running time; in Bocharnikov's research, energy savings were found to be affected by acceleration and braking rates and by running a series of simulations in parallel with a genetic algorithm search method. Chen et al. [7] employed genetic algorithms to optimize train scheduling. The result showed that the method can significantly reduce the maximum traction power. Although these methods and algorithms were effective, they can only be applied in mass rapid transit (MRT) and light rapid transit (LRT) systems. Usually, in MRT, distance between two stations was short and the top running speed was about 80–100 km/h. In this case, a train generally must decelerate in preparation for reaching the next station before it reaches the speed limit. In Milroy's doctoral dissertation [8], *Aspects of Automatic Train Control*, it was proved that for short distance train control represents three different motion regimes, including acceleration, coasting, and braking. But later, in 1984, Howlett [9] proved that in long distance train running, cruising was significant in minimizing energy consumption. Due to the difference between MRT, LRT, and

high-speed trains, these methods cannot be applied in high-speed trains for energy optimization.

For high-speed trains, energy saving and trains control optimization were also studied by scholars. Kawakami [10] from Central Japan Railway Company presents a dynamic power saving strategy for Shinkansen traffic control; the author made conclusion that predictive simulations in every layer and target shooting operation of trains are the basis for energy control. With consideration of track gradient and speed limits, Cheng [11] summarized train control problems with two different models, traction mechanical energy model (TMEM) and traction energy model (TEM), in a long-haul train. Hwang [12] presented an approach to identify a fuzzy control model for determining an economic running pattern for a high-speed railway through an optimal compromise between trip time and energy consumption.

In this paper, taking the Beijing-Shanghai High-Speed Railway as a case, an improved PMPGA was applied to find a perfect running with a specified running. In this research, security, stop precision, and riding comfort were considered and also the railway line parameter includes the slop, tunnel, and curve. The result demonstrates that the PMPGA improved algorithm was better with the SGA and it has achieved conspicuous energy reduction.

## 2. Train Traction Module

**2.1. Train Traction Property.** Traction property curve is an important curve demonstrating the relationship between train traction effort and speed. It was the most significant work when a train was designed. Figure 1 shows the schematic diagram of traction property curve calculation.

In Figure 1, there are three curves; the top one is adhesion-limited braking force  $F_{\max} = f(v)$ , the middle one is traction effort property  $F = f(v)$ , and the bottom one, denoted as  $W$ , is the sum of resistances (e.g., bearing, rolling, air, and grade resistance)  $W = f(v)$ . Note that point A, the cross of  $F_{\max} = f(v)$  and  $W = f(v)$ , correspond  $v_a$ , is greater than the  $v_{\max}$ . Now, according to the curve, traction effort property  $F = f(v)$  could be generated as

$$\begin{aligned} \frac{(F_v - F_{v0})}{v} &= \frac{(F_{v'} - F_{v0})}{v'} & 0 \leq v \leq v' \\ F_v * v &= F_{\max} * v_{\max} & v' \leq v \leq v_{\max}. \end{aligned} \quad (1)$$

In the above formula,  $F_v$  represents the traction force when the speed is  $v$ .  $v'$  is the speed on the intersection point of constant moment segment and constant power segment.  $F_{\max}$  represents traction force limitation.

**2.2. Train Resistance.** To ensure that the TE was able to drive the train with a speed, the total resistances, in this paper, defined as  $W$ , must be known. Total resistances include basic resistance  $W_0$  (axle friction resistance, track resistance, rolling resistance, journal resistance, air force resistance, and vibration resistance) and extra resistance  $W_j$ .  $W_j$  includes grade resistance ( $W_i$ ), curve resistance ( $W_r$ ), and tunnel resistance ( $W_s$ ).

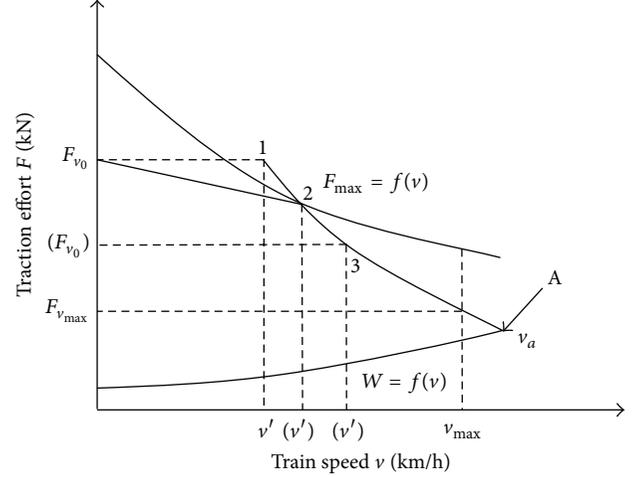


FIGURE 1: Diagram of traction property curve calculation.

It [13] was found that speed was the main factor which effects the basic resistance, and basis resistance can be expressed by a quadratic equation formulated as follows:

$$\omega_0 = a + bv + c \cdot v^2, \quad (2)$$

where the coefficients  $a$ ,  $b$ , and  $c$  are dependent on axle load, number of axles, cross-section of the train, and shape of the train.

According to [14], considering the train as a multiparticle object, we can have the  $w_j(x)$  as the following function:

$$w_j(x) = \frac{1}{L} \left[ \sum i_i * l_i + 600 \sum \frac{l_{ri}}{R} + \sum (w_s * l_s) \right], \quad (3)$$

where  $L$  is the length of the train and  $i_i$  and  $l_i$  represent the gradient and grade length.  $R$ ,  $l_{ri}$  are the curve radius and length.  $w_{si}$ ,  $l_{si}$  are the tunnel resistance and length.

Then, the motion equation and the  $a$ ,  $v_i$ , and  $S_i$  were formulated as below:

$$\begin{aligned} a &= \frac{dv}{dt} = \frac{F - B - (\omega_i + \omega_r + \omega_s + \omega_0)}{M(1 + \gamma)} \\ V_i &= a\Delta t + V_{i-1} \\ S_i &= \frac{V_i + V_{i-1}}{2} \Delta t + S_{i-1}, \end{aligned} \quad (4)$$

where  $V_i$  was the speed of current moment,  $v_{i-1}$  was the speed of last moment,  $a$  was the acceleration of current moment,  $s_i$  was the distance of current moment from the first station, and  $s_{i-1}$  was the distance of the last moment from the first station.

## 3. Traction Energy Module (TEM)

In order to achieve minimal energy consumption, generally, train control for running between stations, including acceleration, cruising, coasting, and braking, should be applied at appropriate time. Golovitcher [15] and Khmel'nitsky [16] analyzed the train movement process with nonlinear constrained

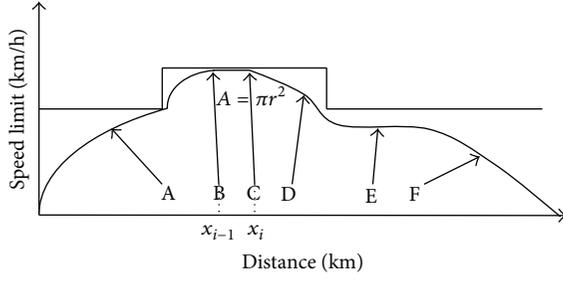


FIGURE 2: Diagram of different running status during one journey.

differential equations and concluded that a maximum economic train running strategy should contain four statuses, maximum traction, cursing, coasting, and maximum braking. For analyzing station-to-station travel time and distance profile, it is essential to comprehend the description of the motion statuses and their mathematical expressions. In maximum traction, power is used to overcome gravity (if climbing) and the dynamic resistance so as to accelerate. When cruising, power is used to overcome the resistance to maintain the train at the constant speed; at this time, the acceleration is zero. When coasting, the running train only suffers from the force of resistance. Applying coasting when the train runs between stations as much as possible is considered to be the most effective energy consumption way. When braking, with regeneration technology fitted, energy can be produced using the motor as a generator.

A train's journey may have variables coast intervals (Figure 2) to achieve an optimal solution. Figure 2 shows a train's status and changing point during a running between two stations. In the figure, the points mean the following: A: traction; B: cursing start point; C: coasting start point; D: coasting; E: cursing; F: braking.

Now, the aim is to find an optimal control strategy for minimal energy consumption in a round trip between two stations. This problem can be seen as a double optimization problem.

Traction energy module can be described as follows.

Make  $X$  the distance between two stations, and travel time was fixed  $T$ ;  $[0, T]$  can be divided as

$$0 = t_0 \leq t_1 \leq t_2 \cdots \leq t_n \leq t_{n+1} = t, \quad (5)$$

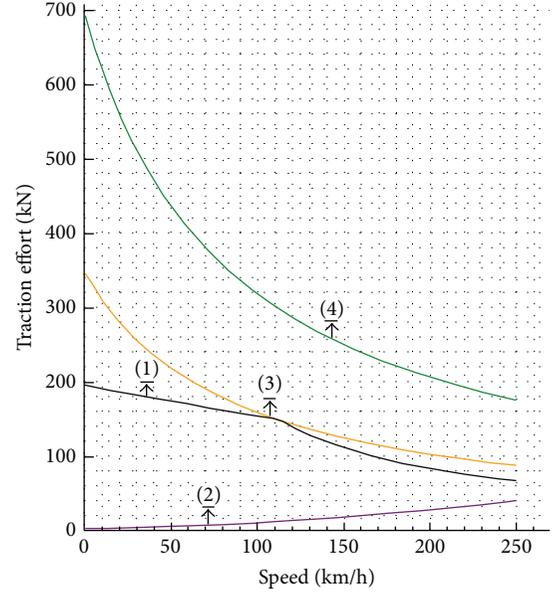
where  $t_0$  is the initial time and  $t_{n+1}$  is the final time; in the time space  $[t_k - t_{k+1}]$  train travel distance is  $[x_k - x_{k+1}]$  and in  $[0, x]$

$$0 = x_0 \leq x_1 \leq x_2 \leq \cdots \leq x_n \leq x_{n+1} = x. \quad (6)$$

Total energy consumed by the train can be defined as follows:

$$\min E = \int_{x_0}^{x_f} u_f(x) f(v) dx$$

$$\text{s.t.} \left\{ \begin{array}{l} \frac{dt}{dx} = \frac{1}{v} \\ v \frac{dv}{dx} = \frac{u_f(x) f(x) - u_b(x) b(v)}{Mg} - w_0(v) - w_j(x) \end{array} \right\}$$



Curve 1: train traction property  
Curve 2: basic resistance  
Curve 3: adhesion-limited braking force (wet)  
Curve 4: adhesion-limited braking force (dry)

FIGURE 3: Train traction property and adhesion-limited braking force.

$$t(x_0) = 0, t(x_f) = T, v(x_0) = 0, v(x_f) = 0$$

$$v \leq V(x), u_f \in [0, 1], u_b \in [0, 1],$$

(7)

where  $E$  is the energy consumption and  $T$  is a fixed time when the train travels between two stations.  $t(x_0)$  is start time,  $t(x_f)$  is arrival time, and  $v(x_0)$  and  $v(x_f)$  represent the start speed and final speed; it was obvious that  $v(x_0)$  and  $v(x_f)$  are equal to 0.  $u_f$  and  $u_b$  were coefficient of traction power and braking.

Then the train control strategy set was  $S = \{s_i\} = \{\text{traction}(T), \text{cursing}(CR), \text{coasting}(C), \text{Braking}(B)\} = \{T, CR, C, B\}$ .

Finally, the train control matrix was defined as

$$C = [c_0, c_1, c_2, \dots, c_i, \dots, c_{n-1}, c_n], \quad (8)$$

where  $c_i = [x_i, s_i]$ ,  $x_i$  is the position, and  $s_i$  is the control strategy start at the position  $x_i$ . From Figure 2, we can see that  $s_i \in S$ .  $x_0 = 0$  and  $x_n$  can be easily calculated by the last braking process.

#### 4. Minimize the Energy Consumption with Parallel Multipopulation Genetic Algorithm

The genetic algorithm (GA) [9, 10] is a method for solving both constrained and unconstrained optimization problems based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions [17]. At each step, the GA

TABLE 1: Experiment of PMPGA with different subpopulation group and gene length.

| Experiment | $N_{sp}$ | Gene length | Group size | $P_c$ | Generation | $P_m$ | $P_v$ |
|------------|----------|-------------|------------|-------|------------|-------|-------|
| E1         | 3        | 50          | 100        | 0.7   | 300        | 0.068 | 0.2   |
| E2         | 3        | 100         | 100        | 0.7   | 300        | 0.068 | 0.2   |
| E3         | 6        | 50          | 100        | 0.7   | 150        | 0.068 | 0.2   |

selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, however, as we all know, the standard GA has the premature convergence phenomenon and slow searching process. In our research, we apply PMPGA, which is a simulation of gene isolation and gene migration in biological evolution process where all populations are divided into many subpopulations with different control. Because the subpopulations have different gene patterns and their genetic processes are independent, the global optimum and the fully search are guaranteed by the difference in evolutionary direction. The optimal individual is quoted by other subpopulations through migration operator.

Finally, considering the optimal object and the constraint conditions, the PMPGA compute process can be described as below.

**4.1. Chromosome.** Take the sequence of train control strategy, which contains the sequence of train operating conditions and the corresponding sequence of conversion locations for the operation section, as a chromosome. The function is

$$C = [(x_0, s_0) (x_1, s_1) \cdots (x_i, s_i) \cdots (x_l, s_l)], \quad (9)$$

where  $c_i = [x_i, s_i]$ ,  $x_i$  is the position, and  $s_i$  is the control strategy start at the position  $x_i$ .  $s_i$  were discrete variables which contain four control strategies [T, CR, C, B]; each control strategy corresponds to one energy consumption formula.  $x_i$  uses the real number encoding.  $l$  is the length of chromosome and it is also variable.

**4.2. Initial Population.** Population is constructed using chromosomes; each chromosome represents a single solution point in the problem space. In our research, donate individual matrix  $U (U \in C)$  was randomly created with different gene length. Gene length means possible times of traction strategy during a running. Consider the distance between two stations. We assign the maximum gene length as  $GL_{max}$ . Each  $U^i$  was a control matrix and all created  $U$ s compose  $N$  subpopulation group; each subpopulation group is denoted as  $P = \{p_1, p_2, p_3, \dots, p_k\}$ , and  $k$  is the number of populations in a subpopulation group. In our research  $N_{sp}$  assigned as number of subpopulation groups will be computed in parallel.

**4.3. Fitness Function.** Applying the individual which means the control matrix to the energy calculated formula, we can get the object value. The fitness evaluation is based on the minimization of the energy consumption, which is defined as

$$\text{Fit}(x) = \frac{1}{C_{max} + \text{obj} + C}. \quad (10)$$

Considering the fastest running strategy, the maximum energy consumption is about 4000 kwh; we make coefficient  $C_{max}$  as 4000 and  $c$  was 0.1.

**4.4. Standard of Convergence.** The convergence criterion is whether the maximum evolutionary generation is reached or the best individual remains unchanged among several generations. If the algorithm is not convergent, then continue to the next operations; otherwise, searching process ends.

Selection operation: Roulette wheel selection first calculates each individual  $x_i'$  corresponding proportion of its fitness value to the total fitness value of the whole population, labeled as  $p_i$ , by

$$p_i = \frac{\text{Fit}(x_i)}{\sum_{j=1}^N \text{Fit}(x_j)}, \quad (11)$$

where  $i = 1, 2, \dots, N$  and  $N$  is the size of population. Then the operator repeats  $N$  times of selecting an individual from the current population to generate the new population. In each time, a random real number  $q$  uniformly scattered in the range  $(0, 1)$  is generated, and the individual  $x_k$  where  $k$  satisfies (20) is selected:

$$k = \min \left\{ j \mid \sum_{i=1}^{j-1} p_i \leq q, j = 1, 2, \dots, N \right\}. \quad (12)$$

It is obvious that, in the roulette wheel selection, the fitter individuals have a greater chance of survival than the greater ones.

**Crossover.** Uniform crossover operator: the crossover operator works as follows. After the two "parents" are drawn, each corresponding pair of coordinates exchanges its values independently, with the same probability  $0 < r < 1$ , as follows:

$$\begin{aligned} X_1^{t+1} &= rX_1^t + (1-r)X_2^t \\ X_2^{t+1} &= rX_2^t + (1-r)X_1^t. \end{aligned} \quad (13)$$

In formula (13)  $X_1^t, X_2^t$  represent the gene of parents and  $X_1^{t+1}, X_2^{t+1}$  represent the next generation.

**Mutation Operator.** Using random number generator to generate a number between 0 and 1, if it is less than the probability of mutation  $p_m$ , chromosomes do mutation. Several mutation positions are rolled randomly.

In order to find the best solution, we define different gene length and different number of subpopulation groups for confrontation. By SGA, the population size is 100, gene

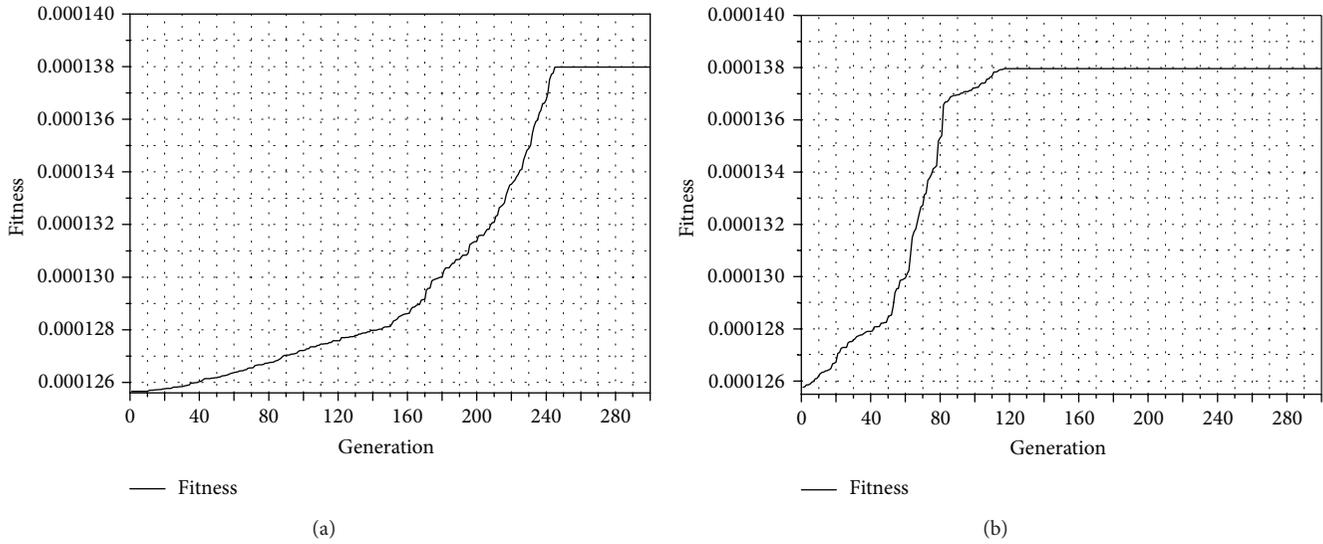


FIGURE 4: Evolutionary curve of standard GA (left) and PMPGA E1 (right).

TABLE 2: Basic train information.

|                                  |           |
|----------------------------------|-----------|
| Motor car/trailer/number of cars | 14/2/16   |
| Number of axles                  | 64        |
| Train weight                     | 895.6 (t) |
| Outpower (kw)                    | 615*16    |
| Voltage rating (v)               | 3000      |
| Current rating (A)               | 230       |
| Highest running speed (km/h)     | 380       |
| Cursing speed (km/h)             | 350       |

length is 50, the maximum evolutionary generation is 300, and  $P_c = 0.7$ ,  $P_m = 0.068$ , and  $P_v = 0.2$ . The specified running time is 20 mins. The adjustment coefficient  $A$  of running performance index function is 3.6. The update time interval is 1 s for multiparticle train simulator (see Table 1).

By PMPGA, we try 3 groups of experiments as below, and the update time interval is 1 s for multiparticle train simulator.

## 5. Case Study and Simulation

In this project, we use *c#* to develop a simulation environment. Then the improved train control strategy can be verified and compared with the previous one. The trains run in the Beijing-Shanghai High-Speed Railway from Beijing to Langfang; the line length is 1305.121 km and the distance between Beijing and Langfang is 59.5 km. Reality line parameters including grade, tunnel, curve, and speed restriction are all considered in the simulation.

Train traction property, basic train information, and reality line parameters were showed in Figure 3, Table 2, and Table 3.

From the simulation result, Figure 4 shows that, with standard GA, the maximum fitness rises much faster after the 140th generation and even faster at the 220th generation; after about the 240th generation, the fitness reaches the maximum

value and becomes stable after that. Compared with the E1, the maximum fitness rises sharply at the 75th generation and becomes stable from the 120th generation. The result shows that the parallel multipopulation GA has the speed of convergence and the precision is considerably improved; also it avoids the premature convergence phenomenon of single-population evolutionary algorithm and maintains the evolutionary stability of the best individuals.

For experiment E1 (Figure 4, right) and E2 (Figure 5, left), we can see that the gene length was extended to 100 which does not cause any improvement. Both curves reach the maximum value and become stable at about the 120th generation. From the result of E1, the gene length 50 is enough for the control strategy between two stations.

For experiment E3, when  $N_{sp}$  was extended from 3 to 6, gene length was set as 50 and generation was set as 150. The speed of convergence was improved. At about the 85th generation, the curves become stable and reach the maximum value.

When applying the control strategy to the simulation system, we got the following result.

From Figure 6 we can see that the running strategy was applied to save energy consumption, and cursing and coasting strategy were also applied in appropriate time. Running results were compared in Table 4.

We can see that when running time from Beijing to Langfang was 16'32" when applying the fastest strategy, energy consumption is 3957.7 kwh. When running time was set extended to 20'00", energy consumption was reduced to about 3252.4 kwh and 3247.2 kwh, which save 17.82% and 17.95% compared with the fastest running time.

In order to verify the efficiency of the PMPGA, we compared it with another optimal algorithm; one is from YanXH who proposed an algorithm based on differential evolution [18] and the other one is from WangDC who proposed a multiobjective fuzzy optimization [19]. We set up module, apply the algorithm at the same train and same

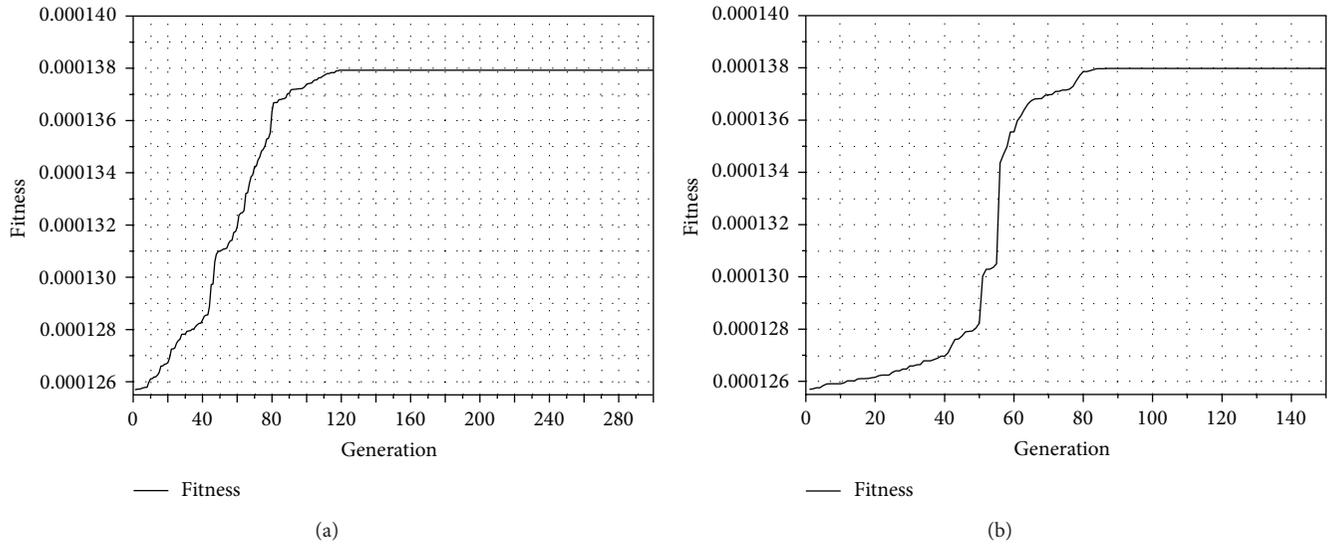


FIGURE 5: Evolutionary curve of E2 (left) and E3 (right).

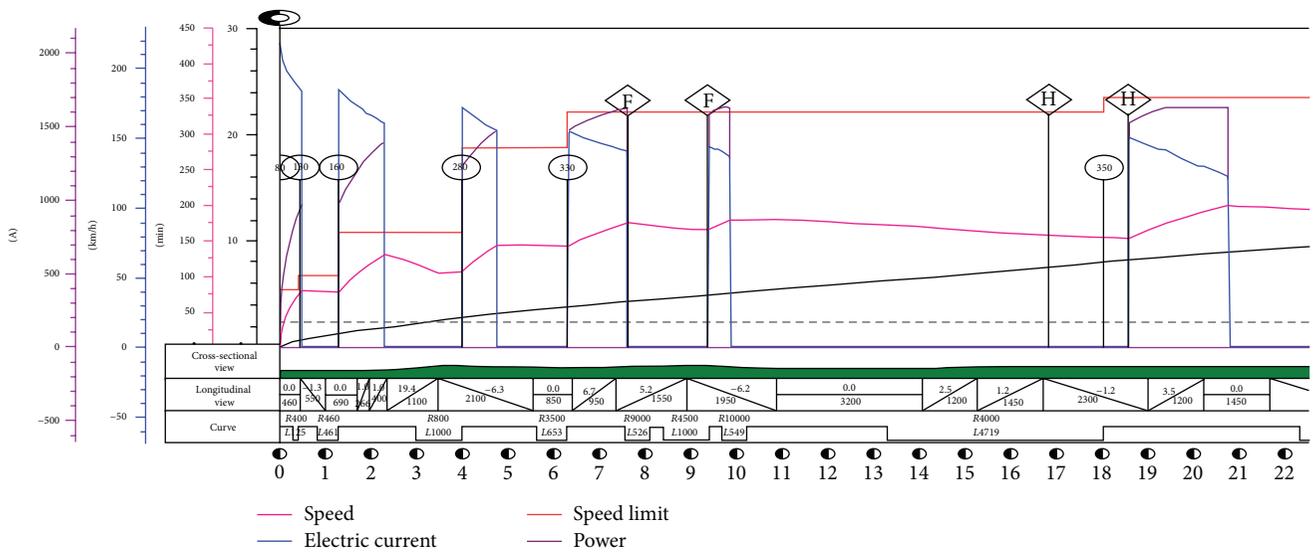


FIGURE 6: Result of normal specified time running strategy.

TABLE 3: Railways line parameters and units.

| Distance | Gradient | Altitude | Slope length | Curve position | Curve radius | Curve length | Station | Speed limit | Tunnel position | Bridge position | Others |
|----------|----------|----------|--------------|----------------|--------------|--------------|---------|-------------|-----------------|-----------------|--------|
| km       | ‰        | m        | m            | km             | m            | m            | km      | km/h        | km              | km              | —      |

TABLE 4: Comparison of running results.

| Rail line                   | Section length | Running strategy       | Time set    | Actual running time | Energy consumption |
|-----------------------------|----------------|------------------------|-------------|---------------------|--------------------|
| Beijing-Langfang            | 59.5 km        | Fastest                | —           | 16 min 32 s         | 3957.7 kwh         |
| Beijing-Langfang with SGA   | 59.5 km        | Specified time         | 20 min 00 s | 19 min 59 s         | 3252.4 kwh         |
| Beijing-Langfang with PMPGA | 59.5 km        | Specified time with GA | 20 min 00 s | 20 min 00 s         | 3247.2 kwh         |

TABLE 5: Experiment confrontation with other algorithms.

| Experiment                  | Section length | Running strategy       | Time set    | Actual running time | Energy consumption |
|-----------------------------|----------------|------------------------|-------------|---------------------|--------------------|
| Beijing-Langfang with PMPGA | 59.5 km        | Specified time with GA | 20 min 00 s | 20 min 00 s         | 3247.2 kwh         |
| Beijing-Langfang E5         | 59.5 km        | Differential evolution | 20 min 00 s | 20 min 00 s         | 3362.9 kwh         |
| Beijing-Langfang E6         | 59.5 km        | Fuzzy optimization     | 20 min 00 s | 19 min 59 s         | 3402.1 kwh         |

railway lines, and get the following results. In Table 5, we define Yan's experiment as E5 and Wang's as E6. The result shows that, with Yan's algorithm, the train was run with a better accuracy in time and E6 is worse. But E5 and E6's experiments show that the energy consumption was about 3.56% and 4.77% more than the PMPGA result. It is proved that the PMPGA algorithm is better with the fuzzy control optimization and algorithm based on differential evolution.

## 6. Conclusion

When a train running schedule is fixed, security, stop precision, and riding comfort must be satisfied. We can save energy consumption by optimizing the control strategy. In this paper, a SGA and PMPGA were applied to find a perfect running based on a specified time. By taking the Beijing-Shanghai High-Speed Railway (Beijing-Langfang section) as a case, the result demonstrates that the SGA and PMPGA were able to reduce energy consumption, but the improved PMPGA has higher speed to convergence and has achieved conspicuous energy reduction; also, PMPGA has achieved better result compared with the multiobjective fuzzy optimization algorithm and differential evolution based algorithm.

## Conflict of Interests

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

## References

- [1] V. Prakash Bhardwaj and Nitin, "On the minimization of crosstalk conflicts in a destination based modified omega network," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 301–314, 2013.
- [2] J. Hui Chong, C. Kyun Ng, N. Kamariah Noordin, and B. Mohd Ali, "Dynamic transmit antenna shuffling scheme for MIMO wireless communication systems," *Journal of Convergence*, vol. 4, no. 1, 2013.
- [3] S. Masoumi, R. Tabatabaei, M.-R. Feizi-Derakhshi, and K. Tabatabaei, "A new parallel algorithm for frequent pattern mining," *Journal of Computational Intelligence and Electronic Systems*, vol. 2, no. 1, pp. 55–59, 2013.
- [4] H. Kumar Gupta, P. K. Singhal, G. Sharma, and D. Patidar, "Rectenna system design in L-band (1-2 GHz) 1.3 GHz for wireless power transmission," *Journal of Computational Intelligence and Electronic Systems*, vol. 1, no. 2, pp. 149–153, 2012.
- [5] L. Yang, Y. Hu, and L. Sun, "Energy-saving track profile of urban mass transit," *Journal of Tongji University*, vol. 40, no. 2, pp. 235–240, 2012.
- [6] Y. V. Bocharnikov, A. M. Tobias, C. Roberts, S. Hillmansen, and C. J. Goodman, "Optimal driving strategy for traction energy saving on DC suburban railways," *IET Electric Power Applications*, vol. 1, no. 5, pp. 675–682, 2007.
- [7] J.-F. Chen, R.-L. Lin, and Y.-C. Liu, "Optimization of an MRT train schedule: reducing maximum traction power by using genetic algorithms," *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1366–1372, 2005.
- [8] I. P. Milroy, *Aspects of automatic train control [Ph.D. thesis]*, Loughborough University, 1980.
- [9] P. G. Howlett, "Existence of an optimal strategy for the control of a train," School of Mathematics Report #3, University of South Australida, 1988.
- [10] T. Kawakami, "Integration of heterogeneous systems," in *Proceedings of the Fourth International Symposium on Autonomous Decentralized Systems*, pp. 316–322, 1993.
- [11] J.-X. Cheng, "Modeling the energy-saving train control problems with a long-haul train," *Journal of System Simulation*, vol. 11, no. 4, 1999.
- [12] H. S. Hwang, "Control strategy for optimal compromise between trip time and energy consumption in a high-speed railway," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 28, no. 6, pp. 791–802, 1998.
- [13] T. Songbai, "Study on the running resistance of Quasi-high speed passenger trains," *Science of China Railways*, vol. 18, no. 1, 1997.
- [14] Z. Zhongyang and S. Zhongyang, "Analysis of additional resistance calculation considering the length of the train and discuss of the curve additional resistance clause in the Traction Regulations," *Railway Locomotive & Car*, vol. 2, 2000.
- [15] I. Golovitcher, "An analytical method for optimum train control computation," *Izvestiya Vuzov Seriya Electrome Chanica*, no. 3, pp. 59–66, 1986.
- [16] E. Khmel'nitsky, "On an optimal control problem of train operation," *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 45, no. 7, pp. 1257–1266, 2000.
- [17] B. Singh and D. Krishan Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, article 13, 2012.
- [18] X. H. Yan, B. G. Cai, and B. Ning, "Research on multi-objective high-speed train operation optimization based on differential evolution," *Journal of the China Railway Society*, vol. 35, no. 9, 2013.
- [19] D. C. Wang, K. P. Li, and X. Li, "Multi-objective energy-saving train scheduling model based on fuzzy optimization algorithm," *Science Technology and Engineering*, vol. 12, no. 12, 2012.

## Research Article

# Botnet Detection Using Support Vector Machines with Artificial Fish Swarm Algorithm

Kuan-Cheng Lin,<sup>1</sup> Sih-Yang Chen,<sup>1</sup> and Jason C. Hung<sup>2</sup>

<sup>1</sup> Department of Management Information Systems, National Chung Hsing University, Taichung 40227, Taiwan

<sup>2</sup> Department of Information Management, Overseas Chinese University, Taichung 40721, Taiwan

Correspondence should be addressed to Kuan-Cheng Lin; [kuanchenglin@gmail.com](mailto:kuanchenglin@gmail.com)

Received 21 January 2014; Accepted 4 March 2014; Published 29 April 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Kuan-Cheng Lin 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 the advances in Internet technology, the applications of the Internet of Things have become a crucial topic. The number of mobile devices used globally substantially increases daily; therefore, information security concerns are increasingly vital. The botnet virus is a major threat to both personal computers and mobile devices; therefore, a method of botnet feature characterization is proposed in this study. The proposed method is a classified model in which an artificial fish swarm algorithm and a support vector machine are combined. A LAN environment with several computers which has infected by the botnet virus was simulated for testing this model; the packet data of network flow was also collected. The proposed method was used to identify the critical features that determine the pattern of botnet. The experimental results indicated that the method can be used for identifying the essential botnet features and that the performance of the proposed method was superior to that of genetic algorithms.

## 1. Introduction

Because of the advancements and innovations in technology, the applications of the Internet of Things (IoT) [1] are rapidly growing, such as cloud computing [2] and smart phone applications. The IoT is not a new type of technology; it is the extension of existing technologies; for example, tens of thousands of smart phones are connected by Wi-Fi, 3G networks, or radio-frequency identification; therefore, using smartphones is a type of IoT, and the development of IoT will be a major trend in the future.

However, because of the recent information explosion, information security has become a crucial topic, even in relation to the IoT. Botnets [3–6] are a recent major threat; when a computer has been infected by a botnet virus, it still functions normally, but the attacker can control the infected computer to threaten the victim by achieving distributed denial of service (DDoS) [7], sending spam, engaging in phishing, or embezzling personal or company data. Botnets are typically composed of three components: a bot herder, a bot client, and a command and control server. The bot herder is the attacker and the bot client is the victim that

is infected by the botnet virus; the command and control server (C & C) is the control server of a botnet and also a communication tool between a bot herder and a bot client. A bot herder typically uses Internet Relay Chat (IRC) protocol to communicate with the command and control server and a bot client. IRC protocol provides real-time one-on-one or group chat room service through a connection to an IRC server, and every chat room is called a channel. A bot herder uses IRC channels to send specific command codes, which are already determined by the bot herder who sent the virus, to a bot client. When a bot client recognizes the specific command code designed by a bot herder, the bot client achieves the movement according to the received command code.

Because botnet viruses are always changing, in both pattern and attack methods, detecting and protecting against these viruses have become extremely difficult. Most botnet-detecting studies have applied basic Internet virus detection methods such as Honeynet and anomaly-based, signature-based, or machine-learning techniques [8]. The anomaly-based and signature-based methods are the most commonly used. In the anomaly-based method, when the detection system observes that the traffic in the user network exhibits

unusual actions, it determines that the user might be the victim of a botnet virus. The advantage of using the anomaly-based method is that unknown botnets can be detected; the disadvantage is that the rate of misjudgment might be high. In the signature-based method, an unusual packet database is typically built, and when the system detects that the Internet packets of a user conform to the database, the user might be infected by a botnet virus. The advantage of using this method is a high detection rate; however, the database must be frequently updated. Because both these methods possess disadvantages, they were not used in this research; instead, the machine-learning method was adopted for detecting botnet viruses. A method that can be used to detect unknown botnet viruses and has a high detection rate was developed by using feature selection, which was used to identify the critical features of botnet viruses.

Feature selection is used for identifying the critical features of a large amount of multidimensional data and subsequently using those features for analysis. For example, if there are 10 computers in an office and a few of them are infected with an Internet virus, the monthly Internet package data of this office must be collected, which is an extremely large data set because it contains thousands of packet transfer records, and every record has multiple features, such as a host IP address, MAC address, and the protocol type. These data must be analyzed, which subsequently reveals the affected computers as those with several feature anomalies. When the relationship between certain features and viruses is identified, those features must be used with precaution in the future.

This example is an application of feature selection. In a large subset of features, the feature subset most representative or most related to a goal must be identified because although every feature is different, some irrelevant features exist, and certain features are noised or redundant. If all these unnecessary features are considered, the complexity of and space necessary for calculations increase, and the correlation between the feature subset and the goal decreases. Therefore, the purpose of feature selection is to filter unnecessary features and to identify the feature subset that is most related to the goal. Moreover, as the feature number increases, the number of possible relevant feature subsets grows exponentially. When the number of features expands to such a large number that people cannot process it, such problems are called a curse of dimensionality. Conducting a search for all the possible feature subsets involves an excessive amount of time and calculation space, which is not cost-effective; therefore, an efficient and effective optimization algorithm must be used for determining the most suitable feature subset by using limited time and calculation space.

The applications of classification and clustering are widely used in various fields, such as recommendation systems [9], voice communication systems [10], and data mining. Applying feature selection can increase the efficiency of classification and clustering, and increasing classification accuracy and performance through feature selection is imperative. Classification refers to classifying data into appropriate categories. Multiple classification methods can be used, such as a decision tree [11], support vector machine (SVM) [12, 13], or neural network [14, 15]. All these methods are

types of supervised learning. Recently, using an SVM has become increasingly common because SVM can achieve high classification with small training sets [13]. The main purpose of the SVM is to establish an optimal hyperplane to classify data and build a classification model.

The metaheuristic algorithm is widely used in various optimization problems, such as feature selection [16, 17] and schedule management [18]. Various metaheuristic algorithms are inspired by natural mechanisms; for example, genetic algorithms (GAs) [19] were inspired by gene mutation and crossover, and particle swarm optimization [20, 21] was inspired by the movement of flocks of birds. Various metaheuristic algorithms exist, such as cat swarm optimization [22], ant colony optimization [23], and artificial fish swarm algorithm (AFSA) [24], which simulates the foraging of fish swarm.

In [25], the results indicated that the AFSA exhibited excellent performance in function optimization, and the potential of applying the AFSA in optimization problems was also revealed. Furthermore, in [26], the researchers proposed a type of feature selection and back-propagation network for botnet detection; however, using an AFSA combined with an SVM classifier might yield superior performance. In this study, a classified model was proposed combining an AFSA algorithm and an SVM. The proposed method was used to identify the critical features determining the pattern of a botnet. The findings indicated that the proposed method can be used to identify the essential botnet features, accurately classifying botnet detection.

Section 2 introduces the SVM, GA, AFSA, and feature characterization of the botnet virus. Section 3 introduces the proposed botnet detection method, using the SVM and the AFSA. Section 4 presents the experiment results and Section 5 provides a conclusion and suggestions for future studies.

## 2. Background

*2.1. Support Vector Machine.* The SVM was proposed by Cortes and Vapnik [27]. It is a supervised learning model based on structural risk minimization [27] and the Vapnik-Chervonenkis dimension [28]. An SVM is typically applied in machine learning [29] and for solving classification or regression problems; therefore, the main purpose of an SVM is identifying the optimal hyperplane to analyze various classification data. The optimal hyperplane possesses the maximal margin associated with the various classification data, as shown in Figure 1. Two black points and three white points exist on the maximal margin line, which represent two types of classification data; these points are called support vectors.

These support vectors can be used for classifying new data. When the data is not linearly separable, the kernel function must be used to map the data into the Vapnik-Chervonenkis dimensional space. Three types of kernel function ( $\Phi$ ) exist: radial basis functions (RBFs), polynomials, and sigmoids. Using the appropriate kernel function for transforming the data is imperative for increasing the

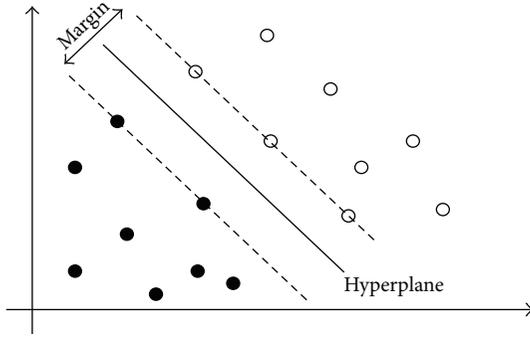


FIGURE 1: The optimal hyperplane.

classification speed. The three kernel functions are described as follows.

RBF kernel:

$$\Phi(x_i - x_j) = \exp(-\gamma \|x_i - x_j\|). \quad (1)$$

Polynomial kernel:

$$\Phi(x_i - x_j) = (1 + x_i \cdot x_j). \quad (2)$$

Sigmoid kernel:

$$\Phi(x_i - x_j) = \tanh(kx_i \cdot x_j - \delta). \quad (3)$$

**2.2. Genetic Algorithm.** The GA was first proposed by J. Holland in 1975, and the main concept of GAs is the simulation of survival of the fittest through crossover and mutation. In this algorithm, chromosomes, which are composed of series genes, play an essential role. Every chromosome has its own fitness value, and the chromosomes that contain high fitness values have a high chance of survival. In this study, an SVM classification accuracy value was used as the fitness value. The GA process is outlined as follows.

- (1) *Initialization.* Encode the optimization problem to integrate with GA, create the fitness function and initial  $N$  chromosome randomly, and include the gene and the parameters.
- (2) *Evaluate Fitness.* Use the fitness function to evaluate the fitness of every chromosome.
- (3) *Reproduction.* Determine the reproduction rate of every chromosome based on its fitness value; if the fitness value is high, the reproduction rate is high as well. Use the roulette wheel selection method to select the reproduction chromosomes.
- (4) *Crossover.* Randomly match two chromosomes from the reproduction pool and create a new generation of chromosomes by completing the crossover step by applying one-point crossover based on the probability of crossover rate.
- (5) *Mutation.* Randomly select dimensions to achieve simple mutation based on the probability of mutation rates; this can increase the opportunities of identifying enhanced solutions.
- (6) *Stop the Algorithm If Terminal Criteria Are Satisfied.* If the terminal criteria are satisfied, stop the algorithm and output

the optimal solution. Otherwise, start from (2) for the next iteration until the terminal criteria are satisfied.

### 2.3. Artificial Fish Swarm Algorithm

**2.3.1. Conception.** The AFSA is an optimization algorithm that simulates the behavior of fish swarm, such as foraging and movement. For example, the position of most fish in a pond is typically the position at which the most food can be obtained. The AFSA includes three main steps, which are Follow, Swarm, and Prey. In the AFSA, these three steps are repeated to determine the optimal solution. Similar to other bioinspired algorithms, the AFSA is used to determine the optimal or most satisfactory solution in a limited time by continually searching for possible solutions using a metaheuristic. In the AFSA, the position of every fish is considered a solution, and every solution has a fitness value that is evaluated using the fitness function. The fitness function changes when different goals are established.

**2.3.2. Process.** The  $F_i$  represent fish  $i$ , and  $C_i$  represent the center of  $F_i$  as mentioned in Table 3. The process of the AFSA is outlined as follows.

- (1) *Initialization.* Encode the optimization problem to integrate with AFSA, create the fitness function and initial  $N$  fish randomly, and include the position and parameters.
- (2) *Evaluate Fitness.* Use the fitness function to evaluate the fitness of every fish.
- (3) *Movement of Fish Swarm.* Process the Follow, Swarm, and Prey movements of every fish and determine the optimal solution.

**Follow.** At this step, the  $F_i$  are compared with neighbors based on the optimal fitness value; if the optimal fitness of its neighbor is superior and the crowded degree of this fish is not greater than the maximal crowded degree, then the  $F_i$  moves to the position of the neighbor fish, which indicates that the feature subset of the  $F_i$  is replaced by that of the neighbor fish. This also indicates that the Follow step is completed. If the Follow step fails, then implement Swarm or Follow for the next fish.

**Swarm.** At this step, the  $F_i$  are compared based on the fitness value of their own,  $C_i$ ; if the fitness value of the  $C_i$  is superior and the crowded degree of the  $C_i$  is not greater than the maximal crowded degree, then the  $F_i$  moves to the  $C_i$ ; this indicates that the feature subset of the  $F_i$  is replaced by that of the  $C_i$  and that the Swarm step is completed. If the Swarm step fails, implement Prey or Follow for the next fish.

**Prey.** At this step, the  $F_i$  randomly changes its own feature subset, indicating that if a feature is 0 and it is chosen to change randomly, this feature becomes 1 and the value of the changed features is not greater than what is visible. If the fitness of the changed feature subset is greater than that of the original, then the changed feature subset replaces the original feature subset which indicates that the Prey step

TABLE 1: Features of the botnet dataset.

| Feature number | Feature name    | Feature content                                 |
|----------------|-----------------|-------------------------------------------------|
| $F_1$          | Total_count     | The number of different destination IP address. |
| $F_2$          | Source_count    | The number of different source IP address.      |
| $F_3$          | Port_count      | The number of different port.                   |
| $F_4$          | Low_port        | The lowest port number.                         |
| $F_5$          | High_port       | The highest port number.                        |
| $F_6$          | TCP_count       | The number of different TCP servers             |
| $F_7$          | UDP_count       | The number of different UDP servers             |
| $F_8$          | ICMP_count      | The number of different ICMP servers            |
| $F_9$          | AvgLength       | Average length of packets                       |
| $F_{10}$       | StddevLength    | The standard deviation of packet length.        |
| $F_{11}$       | Time_Regularity | The time regularity of packet sending.          |
| $F_{12}$       | Info_Char       | The ASCII content of packets                    |

is completed. If the Prey step fails, the algorithm repeats this step until the repeated number reaches the maximal try number.

(4) *Stop the Algorithm If Terminal Criteria Are Satisfied.* If the terminal criteria are satisfied, then stop the algorithm and output the optimal solution; otherwise, start from (2) for the next iteration until the terminal criteria are satisfied.

**2.4. Feature Characterization.** To build a botnet detection system, a botnet network data set must be collected. By referencing [26], a local area network (LAN) simulation was built to collect the packet data of network flow; the computers used in this LAN were affected by a botnet virus. The software VirtualBox was used to simulate 10 computers, and the operating systems of those virtual computers included Windows XP, Windows 7, and Linux; subsequently, the computers were connected to the Internet through a Linux router. On these computers, normal user behaviors were simulated, such as playing online games, browsing websites, and watching videos. The packet data of this LAN was collected for 3 weeks, and the packets included the packet between the C & C server and the botnet virus.

Three data sets (Botnet1, Botnet2, and Botnet3) were obtained using various simulated LANs, and each one was infected by a distinct IRC botnet virus. And the duration of each data set was 1 week, the feature number of every data set was 12, and the instances in every data set were 223. The features of each data set, referenced from [26, 30], are shown in Table 1.

Details regarding the features of AvgLength, StddevLength, Time\_Regularity, and Info\_Char are described as follows.

*AvgLength.* This feature is the average length of every packet and is calculated by using (4). The variable  $x$  is the packet length and  $N$  is the total number of packets:

$$\text{AvgLength} = \frac{\text{SUM}(x_i)}{N}. \quad (4)$$

*StddevLength.* This feature is the standard deviation of the average length of every packet and is calculated by using (5). The variable  $x$  is the packet length,  $\mu$  is the average length of every packet, and  $N$  is the total number of packets:

$$\text{StddevLength} = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}. \quad (5)$$

*Time\_Regularity.* Because a bot client typically transmits a status report packet to a bot herder, knowing the transmission time regularity of each packet was necessary. This feature is the transmission time regularity of specific packets. A transmission time regularity counter is defined as  $\gamma$ , and if the total number of packets is  $N$ , then the total number of  $\gamma$  is  $N-1$ , and a set is an array, (i.e.,  $\gamma = \{\gamma_2, \gamma_3, \dots, \gamma_n\}$ ). For example,  $\gamma_2$  is the transmission time counter that counts the packet number, and the interval time is 2 seconds. Subsequently, the frequency array  $\alpha$  and the infrequency array  $\beta$  were defined. The variable  $t$  is a constant value between 0 and 1 which was set as 0.5 in this study. The feature Time\_Regularity is calculated by using (6):

$$\begin{aligned} \gamma_i > \frac{2t \sum \gamma_i}{N}, & \quad \text{then } \alpha_j = \gamma_i, \\ \gamma_i \leq \frac{2t \sum \gamma_i}{N}, & \quad \text{then } \beta_k = \gamma_i, \end{aligned} \quad (6)$$

$$\text{TimeRegularity} = \text{avg}(\alpha) * (\text{avg}(\alpha) - \text{avg}(\beta)).$$

*Info\_Char.* Because the specific command that a bot herder uses to control the computer of a bot client typically contains symbols, determining the weight of the symbols in the packets is necessary. This feature is the American Standard Code for Information Interchange (ASCII) counter, and 95 counters exist; each counter counts the number of times relevant ASCII characters appear in all packets. For example, a counter was defined as  $C$ ; therefore,  $C_{10}$  is the counter that counts the number of times the ASCII number 10 appears, even as a decimal, or with the symbol #. The feature Info\_Char is calculated by using (7):

$$\text{Info\_Char} = \text{Max}(C_i). \quad (7)$$

### 3. The Proposed Method

Both the GA and AFSA are metaheuristic algorithms; however, they employ distinct optimization mechanisms. The GA has demonstrated success in numerous applications, but

```

Random initialize Fish Swarm.
WHILE (is terminal condition reached)
 FOR ($i = 0$; $i < \text{NumFish}$; $i++$)
 Measure fitness for Fish.
 DO step Follow
 IF (Follow Fail) THEN
 DO step Swarm
 IF (Swarm Fail) THEN
 DO step Prey
 END
 END
 END FOR
End WHILE
Output optimal solution.

```

PSEUDOCODE 1: Pseudocode of AFSA.

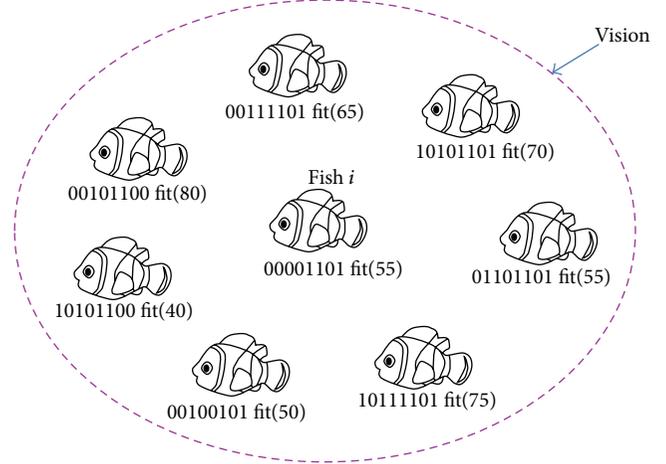


FIGURE 2: Initiation step of AFSA.

TABLE 2: Representation of a solution set.

| $C$ | $\gamma$ | $F_1$ | $\dots F_i \dots$ | $F_n$ |
|-----|----------|-------|-------------------|-------|
|-----|----------|-------|-------------------|-------|

a previous study [25] indicated that AFSA yields superior optimization performance. In this study, the SVM was employed as the classifier, using the AFSA and the GA to perform feature selection. Classifiers can establish a classified model and use it to assign data to the correct categories. First, the data must be divided into multiple components, and every record of this data must have the correct category label. Several pieces of data were regarded as training data and the rest were regarded as test data; subsequently, the training data were input into the classifiers, which was the SVM, to establish the classified model, and then the test data were used to verify this model and obtain accurate classifications. Various components of the data were used to alternately perform these steps, which comprised the cross-validation process. For example, the first portion of the data was used as the test data and the remaining data were used as training data; whereas in the next round, the second portion of the data was used as the test data and the remaining data were used as training data. The pseudocode of AFSA is shown in Pseudocode 1.

In this study, the solution set comprises two parts: (1) the SVM parameters (e.g.,  $C$  and  $\gamma$ ) and (2) the feature subset. In the second part, binary codes were used to represent feature selection; 0 indicated that the feature was not selected and 1 indicated that it was selected. Table 2 shows the solution set.

The feature subset  $F(10101)$  indicates that the first, third, and fifth features were selected, whereas the second and fourth features were not selected. Data input into the SVM without preprocessing indicate that every feature is selected and the classification accuracy is likely unreliable. Thus, the AFSA must be used to conduct feature selection. Incorporating the AFSA with the SVM enables the algorithm to identify a superior feature subset such as  $F(10101)$ . Only data relevant to the selected features are input into the SVM to establish the classification model; this facilitates analyzing whether the classification accuracy is improved. Thus, feature selection is

attained and performing the aforementioned steps enables excluding unnecessary data.

At the initial steps of the AFSA, the algorithm assigns a random feature subset to every fish, and the SVM is used to obtain the classification accuracy based on the fitness of every fish. Subsequently, Follow, Swarm, and Prey processes are implemented to obtain the optimal solution. The definitions of the parameters, referenced from [31], are presented in Table 3.

The steps involved in the AFSA-SVM method are presented as follows.

- (1) Initiation: randomly assign a feature subset to  $N$  fish. Define all parameters including vision, maximal crowded degree, and maximal trial number. For example, Figure 2 shows that eight fish were initiated; each fish has its own feature and the circle represents the vision of fish  $i$ .
- (2) Evaluate the classification value as a fitness value of the feature subset of each fish by using the SVM as shown in Figure 2.
- (3) Starting with the first fish, implement the Follow step. If Follow is successful, perform step 6; otherwise perform step 4. For example, in Figure 2, the fitness value of fish  $i$  is 55; by contrast, the best fitness neighbor exhibits a value of 80. Thus, the best fitness neighbor demonstrates a superior fitness value, indicating that a superior fish is located in the vision of fish  $i$ . Therefore, the Follow step is successful and fish  $i$  moves to the location of the best fitness neighbor, replacing its feature subset as shown in Figure 3.
- (4) Implement the Swarm step for the same fish. If successful, perform step 6; otherwise perform step 5. For example, in Figure 2, calculate the center subset by using (3) in Table 3 and then use the SVM to evaluate its fitness value, comparing the fitness value of fish  $i$  and the center subset. If the fitness value of the center subset is the highest, the Swarm step

TABLE 3: Parameters of AFSA.

| Parameter name             | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Distance                   | The distance between $F_i, F_j$ is obtained through formula (1). Those two fish have the same number of features, $k$ , and if the first feature of $F_i$ is 0 and the first feature of $F_j$ is 0, then the distance between $F_i, F_j$ will remain the same. But if the first feature of $F_i$ is different from the first feature of $F_j$ , the distance between $F_i, F_j$ will be plus one. The distance between two fish is the sum of the differences of every feature:<br>$\text{distance}(F_i, F_j) = \sum_{k=1}^k  F_i(k) - F_j(k)  \quad (1)$ |
| Vision                     | The visibility of a fish and also the maximum distance that this fish can move. In other words, it is the maximum number of features that one fish can change                                                                                                                                                                                                                                                                                                                                                                                             |
| Neighbor                   | The neighbor of $F_i$ is all the fish that are in $F_i$ 's vision; if the distance between $F_k$ and $F_i$ is greater than 0 and less than or equal to vision, $F_k$ is the neighbor of $F_i$ . It is obtained through formula (2):<br>$\text{Neighbor}(F_i) = \{F_k \mid 0 < \text{distance}(F_i, F_k) \leq \text{vision}\} \quad (2)$                                                                                                                                                                                                                   |
| Center                     | The center of $F_i$ is the center of $F_i$ 's neighbor. It can be considered as a fish; the center feature is obtained through formula (3); if more than half $F_i$ 's neighbors' feature $i$ are 0, then the center of $F_i$ 's feature $i$ will be 0, and vice versa:<br>$F_{\text{center}}(i) = \begin{cases} 0, & \sum_{k=1}^k F_k(i) < \frac{k}{2} \\ 1, & \sum_{k=1}^k F_k(i) \geq \frac{k}{2} \end{cases} \quad (3)$                                                                                                                               |
| Crowded degree             | The crowded degree of $F_i$ is to represent the density of $F_i$ 's position; it is obtained through formula (4):<br>$\text{Crowded Degree}(F_i) = \frac{\text{Neighbors of } F_i}{\text{Total number of Fishes}} \quad (4)$                                                                                                                                                                                                                                                                                                                              |
| The maximum crowded degree | The limited number of crowded degree: if the crowded degree of $F_i$ is greater than the limited number, then other fish cannot approach $F_i$ .                                                                                                                                                                                                                                                                                                                                                                                                          |
| The maximum trial number   | The maximum number can perform the Prey movement                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

is successful and fish  $i$  moves to the center subset, replacing the feature subset.

- (5) Implement the Prey step for the same fish. After the Prey step, perform step 6. For example, in Figure 2, the feature subset of fish  $i$  is 00001101. The features randomly change each time the Prey step is executed. The number of changed features must be less than vision and the number of times Prey is executed must be less than the maximal trial number. After changing the feature subset, evaluate the fitness value by using the SVM and compare it with the original feature subset of fish  $i$ ; if the changed feature subset exhibits superior fitness, the Prey step is successful and the feature subset is replaced with the original feature subset.
- (6) Determine if the current fish is the last in the fish swarm. If no, then begin from step 3 and perform the steps for the next fish; if yes, then perform step 7.
- (7) Determine the fitness of every fish; if excellent fitness is observed, then update the optimal solution and perform step 8.
- (8) Determine if the terminal criteria are satisfied and stop the algorithm; otherwise start from step 3 to begin the next iteration. Figure 4 shows the AFSA flow chart.

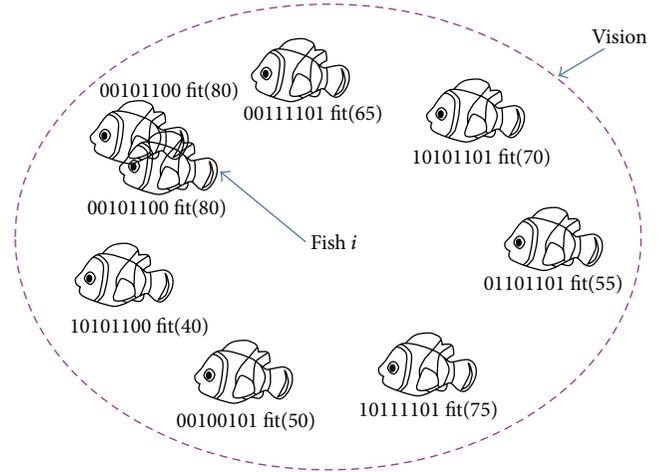


FIGURE 3: Follow step of AFSA.

## 4. Experimental Results

To estimate the performance of feature selection using the AFSA combined with an SVM, the performance of the AFSA was compared with that of a GA, including the classification accuracy, the number of features of the optimal solution subset, and the time spent applying each algorithm to perform calculations. For both the AFSA and GA, the terminal

TABLE 4: The experimental results of AFSA and GA, 5-fold cross-validations.

| Datasets | AFSA-SVM                 |                           |                     | GA-SVM                   |                           |                     |
|----------|--------------------------|---------------------------|---------------------|--------------------------|---------------------------|---------------------|
|          | No. of selected features | Average accuracy rate (%) | Executed time (sec) | No. of selected features | Average accuracy rate (%) | Executed time (sec) |
| Botnet1  | 6                        | 97.76                     | 19843               | 7.2                      | 97.30                     | 22831               |
| Botnet2  | 5.6                      | 98.22                     | 21460               | 6.8                      | 96.87                     | 22868               |
| Botnet3  | 6                        | 99.56                     | 22436               | 7.8                      | 99.11                     | 21583               |

condition of each fold was when the optimal solution was not updated after 1 hour. The algorithm parameters used in this study are presented as follows.

**AFSA.** The number of fish was 30, the maximal number of trials was 30, and the maximal crowded degree was 0.5.

**GA.** The genetic number was 20, and the mutation rate was 0.05.

The computer used to implement the AFSA and GA algorithms was a desktop computer. The operating system was Microsoft Windows 7, the coprocessor was a 2.66-GHz Intel Core 2 Quad Processor Q8400, the amount of memory was 2 GB, and the algorithms were coded using Dev C++. The classifier used was the Library for Support Vector Machines [32] and the RBF kernel function.

**4.1. Experiment 1.** Simulated botnet data sets were collected as mentioned in Section 2.4, and Table 4 shows the experimental results for each data set classified using the AFSA and the GA and a fivefold cross-validation process. The results are the average of the fivefold. The average classification accuracy, number of selected features of the optimal solution subset, and total time between the AFSA and GA were compared. The AFSA was more accurate than the GA was for all data sets, indicating that an increased botnet detection rate can be obtained. The number of selected features of the AFSA was also less than the number of selected features of the GA; thus, the amount of processed data involved in botnet detection was reduced, thereby reducing the detection time. Ultimately, the total time the AFSA spent was less than that of the GA, except for the data set Botnet3; based on these results, the AFSA can be used to obtain higher classification rates, identify the optimal feature subset by using less selected features, and spend less time performing calculations than using the GA can.

To determine the critical features, the total number of selected features in the optimal subset output by using AFSA-SVM was calculated and the results are presented in Table 5. If the number of selected features is high, it indicates that the feature is critical for classifying the input data when using SVM. Thus, the features that exhibit high counts are the features critical to botnet detection.

The results in Table 5 revealed that Features 9 and 11, AvgLength and Time\_Regularity, are the features most often selected from the optimal feature subset, followed by Feature 12, Info\_Char. Because of idle time, the bot herder was not always controlling the computer of the bot client; however, the computer of the bot clients still sent a status report

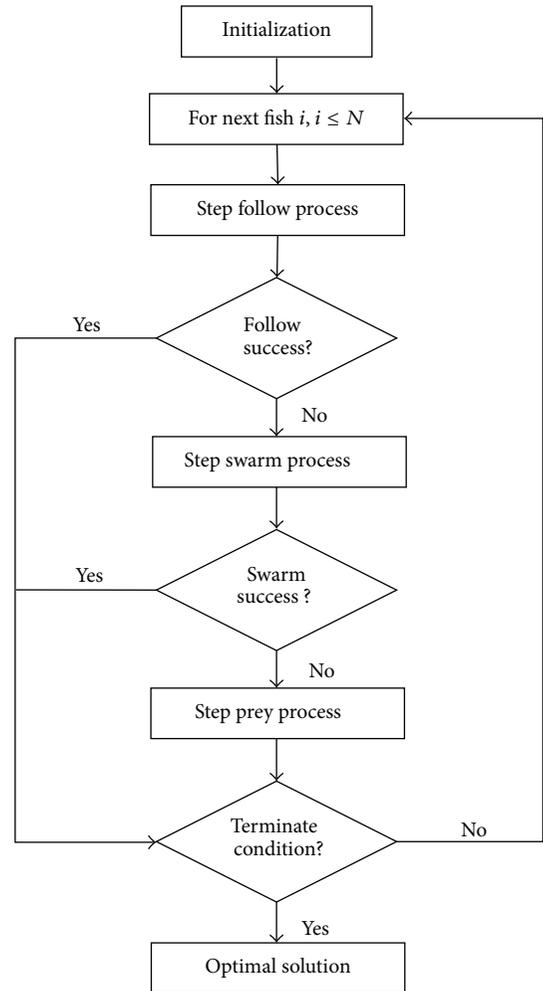


FIGURE 4: Flow chart of the proposed method.

TABLE 5: Count of selected feature by using 5-fold cross-validations.

| Count of selected feature |       |       |       |       |       |       |       |       |          |          |          |
|---------------------------|-------|-------|-------|-------|-------|-------|-------|-------|----------|----------|----------|
| $F_1$                     | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $F_6$ | $F_7$ | $F_8$ | $F_9$ | $F_{10}$ | $F_{11}$ | $F_{12}$ |
| 10                        | 13    | 12    | 12    | 14    | 13    | 10    | 8     | 19    | 16       | 19       | 18       |

packet to the bot herder regularly; therefore, AvgLength is a critical feature. Furthermore, the transmission time interval exhibited a regular pattern in sending the status report packet, which is why Time\_Regularity is such a critical feature. Moreover, because the specific commands sent by the bot herder typically contain specific symbols, identifying the

TABLE 6: The experimental results of AFSA and GA, 10-fold cross-validations.

| Datasets | No. of selected features | AFSA-SVM                  |                     | GA-SVM                   |                           |                     |
|----------|--------------------------|---------------------------|---------------------|--------------------------|---------------------------|---------------------|
|          |                          | Average accuracy rate (%) | Executed time (sec) | No. of selected features | Average accuracy rate (%) | Executed time (sec) |
| Botnet1  | 4                        | 100                       | 3934                | 5.8                      | 97.31                     | 25662               |
| Botnet2  | 4.4                      | 99.11                     | 13505               | 6.2                      | 99.56                     | 5234                |
| Botnet3  | 5                        | 100                       | 10256               | 6.5                      | 97.29                     | 16523               |

TABLE 7: Count of selected feature by using 10-fold cross-validations.

| Count of selected feature |       |       |       |       |       |       |       |       |          |          |          |
|---------------------------|-------|-------|-------|-------|-------|-------|-------|-------|----------|----------|----------|
| $F_1$                     | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $F_6$ | $F_7$ | $F_8$ | $F_9$ | $F_{10}$ | $F_{11}$ | $F_{12}$ |
| 27                        | 27    | 30    | 23    | 32    | 30    | 19    | 23    | 34    | 34       | 39       | 31       |

specific symbols that the bot herder uses may help identify a computer that is infected.

**4.2. Experiment 2.** Tenfold cross-validation was subsequently used, and the terminal condition of each fold was changed as if the optimal solution had not been updated after 1 hour or the classification accuracy was 100%. The results are shown in Table 6. Whether the optimal feature subset falls into the local optimal can be determined. The execution time can be substantially reduced, yielding increased classification accuracy and fewer selected features compared with using fivefold cross-validation. When using the tenfold cross-validation method, the training data grow, enabling the population to comprise additional samples; however, population growth may substantially increase the convergence rate.

The total number of selected features in the optimal subset by using tenfold cross-validations was shown in Table 7. The results shown in Table 7 indicate that Features 9, 10, and 11, representing AvgLength, StddevLength, and Time.Regularity, respectively, were most often selected from the optimal feature subset when using 10-fold cross-validation; this was similar to the results of using fivefold cross-validation, excepting Feature 10 (StddevLength). The classification rate increased when the selected number of StddevLength increased. Therefore, the StddevLength feature was critical to botnet detection. StddevLength represented the standard deviation of the packet length number; the bot clients regularly sent status report packets to the bot herder. These packets were typically short and consistent in length; thus, the StddevLength was the vital feature in botnet detection.

## 5. Conclusion and Future Work

In this study, a feature selection method for detecting botnet viruses is proposed, which is the AFSA-SVM method. Based on the experimental results, using the AFSA yielded only slightly higher classification accuracies than using the GA, but less time was spent to obtain a lesser number of feature subsets. In practical applications, classification accuracy is typically the first priority, but in certain processes, such

as botnet virus detection, detection speed is as crucial as accuracy. To obtain the desired detection speed, the data required for processing must be reduced under the premise that the accuracy level is the same; therefore, in this scenario, the AFSA-SVM method is superior.

The result also shows that both GA and AFSA can still be applied for identifying the critical features of botnet, filtering unnecessary features, and using these algorithms in various applications easily. In our research, an IRC botnet was collected as the data set; however, in real world situations, botnet viruses are constantly changing, and an increasing number of botnet viruses are using peer to peer (P2P) or other protocols as the attack method. Therefore, in future studies, the proposed method must be tested for detecting P2P protocols or other types of botnet viruses. Finally, a feature-selection-based detection system for detecting botnet viruses can hopefully be constructed in the future.

## Conflict of Interests

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

## References

- [1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [2] Y. Pan and J. Zhang, "Parallel programming on cloud computing platforms—challenges and solutions," *Journal of Convergence*, vol. 3, no. 4, pp. 23–28, 2012.
- [3] K. Wang, C.-Y. Huang, S.-J. Lin, and Y.-D. Lin, "A fuzzy pattern-based filtering algorithm for botnet detection," *Computer Networks*, vol. 55, no. 15, pp. 3275–3286, 2011.
- [4] H. Choi and H. Lee, "Identifying botnets by capturing group activities in DNS traffic," *Computer Networks*, vol. 56, no. 1, pp. 20–33, 2012.
- [5] W. T. Strayer, D. Lapsely, R. Walsh, and C. Livadas, "Botnet detection based on network behavior," *Advances in Information Security*, vol. 36, pp. 1–24, 2008.
- [6] M. Abu Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "A multifaceted approach to understanding the botnet phenomenon," in *Proceedings of the 6th ACM SIGCOMM on Internet Measurement Conference (IMC '06)*, pp. 41–52, October 2006.
- [7] M. S. Obaidat and F. Zarai, "Novel algorithm for secured mobility and IP traceability for WLAN networks," *Journal of Convergence*, vol. 3, no. 2, pp. 1–8, 2012.
- [8] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection," in *Proceedings of the 3rd International Conference on Emerging Security Information, Systems and Technologies (SECURWARE '09)*, pp. 268–273, June 2009.

- [9] R. Pan, G. Xu, B. Fu, P. Dolog, Z. Wang, and M. Leginus, "Improving recommendations by the clustering of tag neighbours," *Journal of Convergence*, vol. 3, no. 1, pp. 13–20, 2012.
- [10] A. Bhattacharya, W. Wu, and Z. Yang, "Quality of experience evaluation of voice communication: an affect-based approach," *Human-Centric Computing and Information Sciences*, vol. 2, article 7, 2012.
- [11] J. R. Quinlan, *C4.5: Programs for Machine Learning*, The Morgan Kaufmann Series in Machine Learning, Morgan Kaufmann Publishers, San Mateo, Calif, USA, 1993.
- [12] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, vol. 16, no. 10, pp. 906–914, 2000.
- [13] M. Abdel Fattah, "The use of MSVM and HMM for sentence alignment," *Journal of Information Processing Systems*, vol. 8, no. 2, 2012.
- [14] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 30, no. 4, pp. 451–462, 2000.
- [15] K. Sarkar, M. Nasipuri, and S. Ghose, "Machine learning based keyphrase extraction: comparing decision trees, naïve Bayes, and artificial neural networks," *Journal of Information Processing Systems*, vol. 8, no. 4, pp. 693–712, 2012.
- [16] M. Dash and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 1–4, pp. 131–156, 1997.
- [17] A. James and S. Dimitrijevic, "Ranked selection of nearest discriminating features," *Human-Centric Computing and Information Sciences*, vol. 2, article 12, 2012.
- [18] S. Farzi, "Efficient job scheduling in grid computing with modified artificial fish swarm algorithm," *International Journal of Computer Theory and Engineering*, vol. 1, no. 1, pp. 13–18, 2009.
- [19] C.-L. Huang and C.-J. Wang, "A GA-based feature selection and parameters optimization for support vector machines," *Expert Systems with Applications*, vol. 31, no. 2, pp. 231–240, 2006.
- [20] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.
- [21] B. Singh and D. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, article 13, 2012.
- [22] K.-C. Lin and H.-Y. Chien, "CSO-based feature selection and parameter optimization for support vector machine," in *Proceedings of the Joint Conferences on Pervasive Computing (JCPC '09)*, pp. 783–788, December 2009.
- [23] M. Dorigo, V. Maniezzo, and A. Colnari, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 26, no. 1, pp. 29–41, 1996.
- [24] X.-L. Li, Z.-J. Shao, and J.-X. Qian, "Optimizing method based on autonomous animats: fish-swarm Algorithm," *System Engineering Theory and Practice*, vol. 22, no. 11, pp. 32–38, 2002.
- [25] H. Chen, S. Wang, J. Li, and Y. Li, "A hybrid of artificial fish swarm algorithm and particle swarm optimization for feedforward neural network training," in *Proceedings of the International Conference on Intelligent Systems and Knowledge Engineering*, 2007.
- [26] J. L. Liao and K. C. Lin, *A Study of Feature Selection Integrated with Back-Propagation Network for Botnet Detection*, National Chung Hsing University, Taichung, Taiwan, 2013.
- [27] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [28] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.
- [29] R. Malhotra and A. Jain, "Fault prediction using statistical and machine learning methods for improving software quality," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 241–262, 2012.
- [30] C. Langin, H. Zhou, S. Rahimi, B. Gupta, M. Zargham, and M. R. Sayeh, "A self-organizing map and its modeling for discovering malignant network traffic," in *Proceedings of the IEEE Symposium on Computational Intelligence in Cyber Security (CICS '09)*, pp. 122–129, Nashville, Tenn, USA, April 2009.
- [31] T. Liu, Y.-B. Hou, A.-L. Qi, and X.-T. Chang, "Feature optimization based on Artificial Fish-swarm Algorithm in intrusion detections," in *Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC '09)*, pp. 542–545, April 2009.
- [32] C. C. Chang and C. J. Lin, "LIBSVM: A Library for Support Vector Machines," <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>.

## Research Article

# Adaptive Failure Identification for Healthcare Risk Analysis and Its Application on E-Healthcare

**Kuo-Chung Chu and Lun-Ping Hung**

*Department of Information Management, National Taipei University of Nursing and Health Sciences, No. 365, Mingde Road, Beitou District, Taipei City 11219, Taiwan*

Correspondence should be addressed to Kuo-Chung Chu; [kcchu@ntunhs.edu.tw](mailto:kcchu@ntunhs.edu.tw)

Received 20 January 2014; Accepted 4 March 2014; Published 16 April 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 K.-C. Chu and L.-P. Hung. 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.

To satisfy the requirement for diverse risk preferences, we propose a generic risk priority number (GRPN) function that assigns a risk weight to each parameter such that they represent individual organization/department/process preferences for the parameters. This research applies GRPN function-based model to differentiate the types of risk, and primary data are generated through simulation. We also conduct sensitivity analysis on correlation and regression to compare it with the traditional RPN (TRPN). The proposed model outperforms the TRPN model and provides a practical, effective, and adaptive method for risk evaluation. In particular, the defined GRPN function offers a new method to prioritize failure modes in failure mode and effect analysis (FMEA). The different risk preferences considered in the healthcare example show that the modified FMEA model can take into account the various risk factors and prioritize failure modes more accurately. In addition, the model also can apply to a generic e-healthcare service environment with a hierarchical architecture.

## 1. Introduction

With the trend of information overload, humans face more and more challenges in their activities and have to deal with them [1, 2]. Although most industries incorporate automation techniques into production processes to deal with the challenges, the risk of failure always exists in processes where humans are involved. Moreover, in many industries, such as the aerospace, automobile, and healthcare sectors, human safety is the primary concern; hence, risk management is a hot topic in such industries. Strategies used to manage risk include transferring the risk to another party, avoiding the risk, reducing the negative effect of the risk, and accepting some or all of the consequences of a particular risk.

Certain aspects of many risk management standards have been criticized because they do not achieve a measurable reduction in risk, even though confidence in the estimates and decisions based on the standards is raised. Risk management can be defined as the identification, assessment, and prioritization of risks followed by the coordinated and economical application of resources to minimize, monitor,

and control the probability and/or impact of unfortunate events [3] or to maximize the realization of opportunities. Various industries (e.g., manufacturing and aviation) have long used this risk assessment process to evaluate system safety, and healthcare organizations are now using it to evaluate and improve the safety of patient care services. The risk management field is no different from any other area of management where standards proliferate. It is necessary to highlight some of the most important terms used in the field of risk management and provide examples of how they are defined in some of the well-known reference materials. The ISO Guide 73:2009 [4] defines the terms used in risk management. Its objective is “to encourage a mutual and consistent understanding of, and a coherent approach to, the description of activities relating to the management of risk, and the use of uniform risk management terminology in processes and frameworks dealing with the management of risk.” The first edition of the ISO/IEC Guide 73 was published by the ISO Technical Management Board (TMB) Working Group 2 on risk management terminology. The second edition was compiled by the ISO TMB Working Group on

risk management in association with the development of ISO 31000 to reflect changes in risk management practices and feedback from users.

Studies of safety in the healthcare and other sociotechnological industries have demonstrated repeatedly that human error is the cause of many accidents in complex systems. In air traffic control, for example, it has been found that 80–90% of accidents are caused by human error rather than technical malfunctions [5]. The statistics for healthcare services are similar. For example, in [4], it was reported that 82% of anesthesia-related accidents were due to human error. The causes of human failure in the healthcare industry are the same as those in other industries, for example, distractions, mental fatigue, misdirected attention, and misinterpretation of information [6]. A 1999 report published by the American Hospitals Association estimated that at least 44,000, and perhaps as many as 98,000 Americans, die every year due to errors made in hospitals [7]. The figure is higher than the number of people who die annually in the United States as a result of motor vehicle accidents (43,458), breast cancer (42,297), or AIDS (16,516) [8]. If we evaluate the human tragedy in terms of financial costs, medical errors rank among the most urgent and widespread public problems. The Institute of Medicine (IOM) report [8] on the quality of healthcare in America (entitled *To Err is Human: Building a Safer Health System*) states that “. . . healthcare is a decade or more behind other high-risk industries in its attention to ensuring basic safety.” Thus, we must pay more attention to healthcare industry that depends on perfect human performance and endeavor to eliminate adverse events and medical errors in the industry [9, 10].

To provide safe healthcare services, the industry must use every possible means to reduce risks. Generally, human errors are unavoidable because they are caused by environmental factors rather than incompetence on the part of the individuals involved. It is necessary to enhance patient care practices and establish standard operating procedures (SOPs). In [8], the authors posit that human errors occur because good people have to work in bad systems that need to be made safer [11]. Improving service quality and risk management may improve patient safety.

Failure mode and effect analysis (FMEA) is a technique that identifies the potential failure modes of a product or a process, determines the effects of failures, and assesses the criticality of the effects on the functionality of the product or service. It provides a mechanism for reliability prediction and process design. According to BS 5760 Part 5 [12], “FMEA is a method of reliability analysis intended to identify failures, which have consequences affecting the functioning of a system within the limits of a given application, thus enabling priorities for action to be set.” It has been shown that FMEA is a useful tool for identifying potential failures in a tabular and structured manner. In an FMEA table, a list of critical items helps individuals identify potential failures and ensure the safety of the operating procedures.

However, the risk priority number (RPN) defined in FMEA cannot identify some failures. This shortcoming is due to the nonlinear structure of the RPN function in which the three parameters, that is, severity, occurrence, and

detectability (SOD), are equally important. The RPN function has difficulty differentiating the type of risk (i.e., the failure mode). In an attempt to resolve the problem, we propose a generic RPN (GRPN) function that assigns a weight to each parameter so that the weights represent individual industry preferences for the parameters. The function is calculated with the logarithm of the weight and then transformed into a linear function to estimate the risk independently of the three parameters. The GRPN function-based FMEA model is capable of differentiating the type of risk, and it satisfies the requirement for diversified risk preferences. To validate the proposed adaptive risk identification model, we apply it to a case of testing Down syndrome and compare the results with those derived using the traditional RPN approach.

In addition, the global population is predicted to expand with both a shrinking number of economically active and a larger proportion of older people. The number of people with long-term conditions will increase the importance of perceived health. Due to the constant advances of mobile and wireless technologies, user-generated service is a development trend of mobile services [13]. Using the technologies to improve people’s health and the delivery of healthcare have not only brought about caregiver/care provider connectivity but have brought the healthcare into a new era of ubiquitous/pervasive healthcare [14–16]; there are several examples: stroke patient monitoring and guidance for promoting rehabilitation, location tracking, vital signs and well-being data acquisition and analysis, fall detection, behavior tracking, and sleep analysis. No matter what the examples are, a lot of sensing devices are involved in distributed environment that requires a collaborative decision analysis system or workflow-driven healthcare platform for collaborative applications [17]. To facilitate the ubiquitous service, an ontology-based evaluation model is proposed to ensure the service quality [18]; while an emerging area called intelligent environments provide an integrated approach for collaborative data management of ubiquitous services [19]. Those studies show that e-healthcare is an emerging research issue and thus we propose the application of adaptive risk identification model on e-healthcare.

The remainder of this paper is organized as follows. In Section 2, we review the literature on risk management and the FMEA model. In Section 3, we propose a modified FMEA model called GRPN that includes model formulation, validation, and simulation. In Section 4, we conduct sensitivity analysis to compare the model’s performance with that of RPN. In Section 5, we present a case study of healthcare risk analysis and show the adaptability of the proposed approach; in Section 6, we also apply the proposed model to e-healthcare environment; in Section 7, we conclude this paper with contributions and discussions.

## 2. Related Work

**2.1. Failure Model and Effect Analysis (FMEA).** FMEA has been used in the aerospace and automobile industries for several decades. The aerospace industry used FMEA as a formal design methodology in the 1960s because of the need for a

TABLE 1: Description of the three risk factor scales.

| Scale | Factors              |                |                                           |
|-------|----------------------|----------------|-------------------------------------------|
|       | Severity             | Occurrence     | Detectability                             |
| 1~2   | Insignificant effect | Rare           | Will detect a failure                     |
| 3~4   | Minor effect         | Unlikely       | Likely to detect a failure                |
| 5~6   | Moderate effect      | Possible       | Might detect a failure                    |
| 7~8   | Major effect         | Likely         | Unlikely to detect a failure              |
| 9~10  | Hazardous effect     | Almost certain | Detection of a failure is highly unlikely |

high level of reliability and safety. It is now used extensively to ensure the safety and reliability of products/processes in a wide range of industries, particularly the aerospace, automotive, and nuclear industries. In FMEA, the RPN is used to assess the level of risk based on three factors. The Potential Failure Mode and Effects Analysis Manual [20], section QS-9000, classifies the risk factors as follows: (1) severity (*S*): a rating of the seriousness of the effects of a potential failure; (2) occurrence (*O*): a rating of the likelihood that the failure will occur; (3) detectability (*D*): a rating of the likelihood that the current detection methods or controls will detect a potential failure mode. The three factors are rated on a scale of 1 to 10 on the basis of degree, as shown in Table 1. The RPN, which is denoted as a traditional RPN (TRPN), is the product of severity, occurrence, and detectability, as expressed in

$$\text{TRPN} = S \times O \times D. \quad (1)$$

The TRPN provides the foundation for improvement; that is, the larger the TRPN, the greater the potential for improvement. Corrective action is taken by the relevant departments, beginning with the department that makes the largest contribution to the risk. After corrections are made, the TRPN should be recalculated to determine if the risks have been reduced and to check the effectiveness of the corrective actions taken by each contributor.

To begin with FMEA, a high-level process flowchart should be compiled and appropriate knowledge resource experts should be selected to form an FMEA project team. An FMEA knowledge expert should be nominated to train team members in the selected process. On completion of their training, the team should start to build an FMEA model for the process. From the high-level flowchart, the team should identify the process functions and determine the scope of the project.

There are five steps in the FMEA method:

- (1) select a procedure/subprocedure for study;
- (2) assemble a team;
- (3) make a diagram of the procedure/subprocedure;
- (4) identify the failure modes (risks):
  - (a) brainstorm potential failure modes, ascertain why they might happen, and determine their

effects in terms of the occurrence, severity, and detectability criteria;

- (b) compile a worksheet for risk analysis, and rank the risk for each failure point;

- (5) take corrective action:

- (a) redesign the process if the effects of errors are unacceptable;
- (b) analyze, test, implement, and monitor the new process.

*2.2. Application of FMEA in Different Industries.* The FMEA tool was developed by the US military in the late 1940s to evaluate system and equipment failures. Since then, it has been widely used in various industries. For example, the aerospace industry began utilizing FMEA in the mid-1960s, and it was adopted by the healthcare industry in the late 1990s. FMEA helps healthcare organizations reduce potential risks and allows them to develop control strategies for high-risk processes. In hospitals, for example, improving service quality and risk management to ensure patient safety are becoming increasingly critical. The Joint Commission (TJC) standard LD.4.40 regards proactive risk assessment as an element of the performance of all accredited facilities. Since 2003, TJC has mandated all accredited organizations to analyze at least one high-risk process annually and identify ways that a breakdown or process failure could occur. Organizations are also required to prioritize potential process breakdowns, redesign the processes, and assess the effects of any changes that are made [21].

FMEA is exactly the type of technique or model that TJC recommends to fulfill all of the above requirements. Like any new strategy, refining an FMEA model takes some practice; however, once the model is established, it becomes an indispensable technique in any hospital's risk assessment plan. In response to public concern about medical errors, the Joint Commission on Accreditation of Healthcare Organizations (JCAHO) promised to enhance patient safety. Since 1996, JCAHO has introduced several standards to improve patient safety; and it set October 2001 as the date that all healthcare facilities had to have some kind of risk assessment framework in place. The commission did not specify the process that had to be used; however, the FMEA model satisfies the requirement. Under the JCAHO directive, facilities must perform a proactive risk assessment of at least one high-risk process annually. The choice of process can be driven by internal patient safety needs or the JCAHO sentinel event alerts. Much of what needs to be done to improve safety in the healthcare sector has been accomplished already in other industries. In 2001, the JCAHO chose the FMEA as an appropriate safety improvement technique for healthcare services.

*2.3. Critique of the TRPN Model.* Since more than 40 years, FMEA has been used successfully in various industries to predict how a work process may fail or how a device may be used incorrectly [22]. FMEA involves close examination

of high-risk procedures or error-prone processes to identify improvements that would reduce the occurrence of unintended adverse events. The method provides a straightforward, proactive process of risk identification and quality improvement that is simple to learn and is applicable in all settings. FMEA has proven to be one of the most important proactive measures that can be adopted to prevent failures and errors from occurring in a system, design, process, or service so that they do not reach the customer. However, for various reasons, the TRPNs have attracted a considerable amount of criticism [23–28].

The shortcomings of the TRPN model are analyzed in depth [29]. Here, we review them briefly. Recall that the TRPN is the mathematical product of three factors ( $S$ : the severity of the effect,  $O$ : the probability of occurrence, and  $D$ : the probability of detectability) related to a failure mode rated on a scale of 1 to 10 based on a number of linguistic terms. The first shortcoming is that the TRPN elements are not weighted equally in terms of risk. As a result, SOD scenarios in which their TRPNs are lower than other combinations could still be dangerous. For example, in a scenario with very high severity, a low rate of occurrence, and very high detectability, the TRPN  $9 \times 3 \times 2 = 54$  is lower than in the scenario with a moderate severity level, moderate rate of occurrence, and low detectability where the TRPN  $4 \times 5 \times 6 = 120$ , even though it should have a higher priority for corrective action. The second shortcoming is that the TRPN scale has some nonintuitive statistical properties. The initial and correct assumption that the scale starts at 1 and ends at 1000 often leads to incorrect assumptions about the midpoint of the scale. The 1000 TRPN numbers are generated from all possible combinations. However, most TRPN values are not unique, and some are recycled up to 24 times.

### 3. Method

#### 3.1. Model Formulation

**3.1.1. Formulation of a Generic RPN (GRPN).** The traditional FMEA model assumes that the contributions of the risk factors (SOD) to the value of the TRPN are homogeneous. However, the importance of each indicator probably depends on the type of industry. Therefore, a modified RPN function is needed to provide a more generalized application and to rectify any bias in the TRPN indicators. For example, in aerospace, automotive, and medical applications, the impact of severity ( $S$ ) on the failure effect should be greater than that of frequency ( $O$ ). When failures occur, regardless of the frequency, high priority should be given to taking corrective action. Because the above-mentioned industries involve the safety of people, the importance of  $S$  is significantly greater than that of  $O$ . By contrast, in commodity manufacturing, the priority is to reduce the frequency ( $O$ ) of failures, so  $O$  is more important than  $S$ . To differentiate between the priorities of the three TRPN indicators (SOD), we denote their weights as  $w_S$ ,  $w_O$ , and  $w_D$ , respectively; the weight is an exponent of the indicator, such that  $w_S + w_O + w_D = 1$ . Then, the expression of the logarithm operation represents the RPN as

TABLE 2: Priority of the risk weights with respect to concern priority of the risk factors.

| Concern priority of the risk factors | Priority of the risk weights  |
|--------------------------------------|-------------------------------|
| $S > O > D$                          | $w_S (H) > w_O (M) > w_D (L)$ |
| $S > D > O$                          | $w_S (H) > w_D (M) > w_O (L)$ |
| $O > S > D$                          | $w_O (H) > w_S (M) > w_D (L)$ |
| $O > D > S$                          | $w_O (H) > w_D (M) > w_S (L)$ |
| $D > S > O$                          | $w_D (H) > w_S (M) > w_O (L)$ |
| $D > O > S$                          | $w_D (H) > w_O (M) > w_S (L)$ |

a linear function of the parameters. We define the function as a generic RPN (GRPN) with two types of parameters, namely, risk factors and risk weights, as expressed in

$$\begin{aligned} \text{GRPN}(w_S, w_O, w_D) &= \log(S^{w_S} \cdot O^{w_O} \cdot D^{w_D}) \\ &= w_S \log S + w_O \log O + w_D \log D. \end{aligned} \quad (2)$$

**3.1.2. Using a GRPN-Based FMEA Model.** The GRPN, which is a modified FMEA model, is a function of the risk factors (SOD) and the weights ( $w_S$ ,  $w_O$ ,  $w_D$ ). Although the failure model is related to  $S$ ,  $O$ , and  $D$ , the three factors are independent; therefore, each of them can be described by a stochastic model. To apply the modified FMEA model based on the GRPN function, we consider the possible effects of the factors and the values of the weights.

- (i) Risk factors (SOD): to evaluate the feasibility of the modified FMEA, SOD can be simulated as a stochastic model, for example, with a uniform (U) distribution or a normal (N) distribution. The SOD factors form eight combinations: UUU, UUN, UNU, UNN, NUU, NUN, NNU, and NNN.
- (ii) Risk weights ( $w_S$ ,  $w_O$ ,  $w_D$ ): the weight of each factor is given a value, that is, low (L), medium (M), or high (H). The weights form six combinations: LMH, LHM, MLH, MHL, HLM, and HML. In fact, the weight combinations could vary in different organizations. The weights can be arbitrarily assigned only if the sum of the weights is equal to 1 ( $L + M + H = 1$ ). For example, if the factor  $S$  is more important than the factor  $O$ , it will give a larger value of the weight  $w_S$  than the value of the weight  $w_O$ , and vice versa. On the basis of concern priority of the risk factors, we illustrate possible weight priority respective to the risk factors, as in Table 2. In this paper, for example, we give  $L = 0.1$ ,  $M = 0.3$ , and  $H = 0.6$ . In addition, we consider a special weight (E, E, E), where  $E = 0.333$  (1/3), to be equivalent to TRPN-based FMEA model.

Both the factor distributions and the weights consist of 56 combinations. To determine the applicability of the proposed model, we assess the effect of the GRPN values in all combinations of the parameters. For all the 56 combinations, let  $D_i$  denote the  $i$ th distribution of the GRPN values, and let  $T_i$  denote the acceptable level of the risk value

TABLE 3: Possible combination of the risk factors and weights.

| GRPN<br>( $D_i, T_i$ ) | Combination of ( $w_S, w_O, w_D$ ), GRPN |                      |                      |                      |                      |                      | TRPN                 |           |
|------------------------|------------------------------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|-----------|
|                        | (L, M, H)                                | (L, H, M)            | (M, L, H)            | (M, H, L)            | (H, L, M)            | (H, M, L)            |                      | (E, E, E) |
| (S, O, D)              |                                          |                      |                      |                      |                      |                      |                      |           |
| (UUU)                  | ( $D_1, T_1$ )                           | ( $D_2, T_2$ )       | ( $D_3, T_3$ )       | ( $D_4, T_4$ )       | ( $D_5, T_5$ )       | ( $D_6, T_6$ )       | ( $D_7, T_7$ )       | TRPN1     |
| (UUN)                  | ( $D_8, T_8$ )                           | ( $D_9, T_9$ )       | ( $D_{10}, T_{10}$ ) | ( $D_{11}, T_{11}$ ) | ( $D_{12}, T_{12}$ ) | ( $D_{13}, T_{13}$ ) | ( $D_{14}, T_{14}$ ) | TRPN2     |
| (UNU)                  | ( $D_{15}, T_{15}$ )                     | ( $D_{16}, T_{16}$ ) | ( $D_{17}, T_{17}$ ) | ( $D_{18}, T_{18}$ ) | ( $D_{19}, T_{19}$ ) | ( $D_{20}, T_{20}$ ) | ( $D_{21}, T_{21}$ ) | TRPN3     |
| (UNN)                  | ( $D_{22}, T_{22}$ )                     | ( $D_{23}, T_{23}$ ) | ( $D_{24}, T_{24}$ ) | ( $D_{25}, T_{25}$ ) | ( $D_{26}, T_{26}$ ) | ( $D_{27}, T_{27}$ ) | ( $D_{28}, T_{28}$ ) | TRPN4     |
| (NUU)                  | ( $D_{29}, T_{29}$ )                     | ( $D_{30}, T_{30}$ ) | ( $D_{31}, T_{31}$ ) | ( $D_{32}, T_{32}$ ) | ( $D_{33}, T_{33}$ ) | ( $D_{34}, T_{34}$ ) | ( $D_{35}, T_{35}$ ) | TRPN5     |
| (NUN)                  | ( $D_{36}, T_{36}$ )                     | ( $D_{37}, T_{37}$ ) | ( $D_{38}, T_{38}$ ) | ( $D_{39}, T_{39}$ ) | ( $D_{40}, T_{40}$ ) | ( $D_{41}, T_{41}$ ) | ( $D_{42}, T_{42}$ ) | TRPN6     |
| (NNU)                  | ( $D_{43}, T_{43}$ )                     | ( $D_{44}, T_{44}$ ) | ( $D_{45}, T_{45}$ ) | ( $D_{46}, T_{46}$ ) | ( $D_{47}, T_{47}$ ) | ( $D_{48}, T_{48}$ ) | ( $D_{49}, T_{49}$ ) | TRPN7     |
| (NNN)                  | ( $D_{50}, T_{50}$ )                     | ( $D_{51}, T_{51}$ ) | ( $D_{52}, T_{52}$ ) | ( $D_{53}, T_{53}$ ) | ( $D_{54}, T_{54}$ ) | ( $D_{55}, T_{55}$ ) | ( $D_{56}, T_{56}$ ) | TRPN8     |

TABLE 4: An example of corrective action on the failure mode (FM) with respect to  $T$ .

| Risk weight* |       |       | GRPN | Action or not with respect to given threshold $T^{**}$ |            |            |            |            |
|--------------|-------|-------|------|--------------------------------------------------------|------------|------------|------------|------------|
| $w_S$        | $w_O$ | $w_D$ |      | $T = 0.60$                                             | $T = 0.65$ | $T = 0.70$ | $T = 0.75$ | $T = 0.80$ |
| 0.1          | 0.3   | 0.6   | 0.78 | Yes                                                    | Yes        | Yes        | Yes        | No         |
| 0.1          | 0.6   | 0.3   | 0.72 | Yes                                                    | Yes        | Yes        | No         | No         |
| 0.3          | 0.1   | 0.6   | 0.70 | Yes                                                    | Yes        | Yes        | No         | No         |
| 0.3          | 0.6   | 0.1   | 0.60 | Yes                                                    | No         | No         | No         | No         |
| 0.6          | 0.1   | 0.3   | 0.52 | No                                                     | No         | No         | No         | No         |
| 0.6          | 0.3   | 0.1   | 0.48 | No                                                     | No         | No         | No         | No         |

\* ( $S, O, D$ ) = (2, 5, 8) and  $GRPN = \log(S^{w_S} \cdot O^{w_O} \cdot D^{w_D})$ .

\*\*  $T$  denoted as GRPN threshold  $T_i$ .

(GRPN threshold) for the  $i$ th combination. In addition, for comparison, TRPNs (TRPN1~TRPN8) are also simulated with the risk factors (SOD) in both uniform and normal distributions. The overall combination is shown in Table 3.

Risk analysis is based on an acceptable risk probability  $\alpha$ , which is assigned by organization, department, or process. Given  $\alpha$ , a threshold can be precalculated, where  $\text{Prob}(GRPN \leq T_i) = \alpha$  and the probability that the GRPN value is less than or equal to  $T_i$  is equal to  $\alpha$ , as shown in Figure 1. Whenever the GRPN value  $> T_i$ , priority should be given to take corrective action on the failure mode (FM). The threshold  $T_i$  can be analyzed and suggested by a simulation approach with respective scenarios, as discussed in Section 3.3.

In this section, we give an example to show how the proposed model works. Having an FM, for example, we assign a risk factor ( $S, O, D$ ) to (2,5,8), and the risk weight (L, M, H) is given (0.1, 0.3, 0.6). Corrective action should be taken on the FM whenever its GRPN value is greater than or equal to a given threshold  $T_i$ . The weight combination ( $w_S, w_O, w_D$ ) is illustrated in Table 4 to show whether we should act on the FM. To use the proposed model, we summarize the procedure in the following steps in which  $GRPN'$  and  $GRPN''$  denoted the GRPN values that are calculated in simulated and real environments, respectively. The most important thing is to decide the threshold  $T_i$  by a simulation process with a given  $\alpha$ .

- (1) Use historical data of risk factors ( $S, O, D$ ) to build a probability model of the factors.

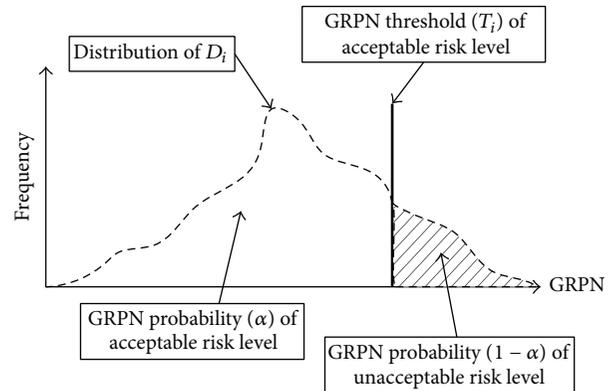


FIGURE 1: Analysis of the acceptable risk.

- (2) Give  $\alpha$  and risk weights ( $w_S, w_O, w_D$ ) according to organization policy.
- (3) Suggest threshold  $T_i$  by analyzing  $GRPN'$  with probability model from step (1), such that  $\text{Prob}(GRPN \leq T_i) = \alpha$ .
- (4) Create an FMEA worksheet (a comprehensive worksheet example will be given in Section 5.1), and compute  $GRPN''$  on all FMs.
- (5) Act on FMs whose  $GRPN''$  are greater than  $T_i$ .
- (6) Repeat steps (2) to (5) until  $GRPN'' \leq T_i$ .

TABLE 5: The statistics of simulation input functions.

| Name              | S (U)    | O (U)    | D (U)    | S (N)    | O (N)    | D (N)    |
|-------------------|----------|----------|----------|----------|----------|----------|
| Min.              | 1.00057  | 1.000374 | 1.000033 | 1.484494 | 1.780332 | 1.560566 |
| Mean              | 5.500004 | 5.5      | 5.499998 | 5.499975 | 5.500002 | 5.500066 |
| Max.              | 9.999767 | 9.999713 | 9.999903 | 9.266962 | 9.414956 | 10.13493 |
| 5% of percentile  | 1.449774 | 1.449551 | 1.449979 | 3.854347 | 3.854757 | 3.854582 |
| 95% of percentile | 9.549781 | 9.54951  | 9.549447 | 7.144094 | 7.144127 | 7.144537 |

3.2. Model Validation—Simulation Approach

3.2.1. Simulation Models and Inputs. The normal distribution is that random variable  $X$  is with the probability density function as defined in (3), where  $\mu$  and  $\sigma$  are mean and standard deviation, respectively. The former determines central tendency, while the latter measures the degree of dispersion. The distribution can be expressed as  $X \sim N(\mu, \sigma)$ , where  $\pi = 3.14159 \dots$  and  $e = 2.71828 \dots$ . Consider

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{(-1/2)((x-\mu)/\sigma)^2}, \quad -\infty < x < \infty. \quad (3)$$

In addition, uniform distribution is that random variable  $Y$  is with the equal probability in the range of  $(a, b)$  as defined in (4), where the probability function value is independent of the variable  $y$ . The distribution can be expressed as  $Y \sim U(a, b)$ :

$$f(y) = \frac{1}{b-a}, \quad a < y < b. \quad (4)$$

The validation uses @RISK decision tool, Palisade Corporation [30]. On the basis of simulation settings, the simulation models by way of @RISK functions, RISKUNIFORM (I, 10) and RISKNORM (5.5, 1), are defined as follows, where C3, C4, K3, K4 are names of cells in a datasheet of Microsoft Excel, and RiskStatic () is a function of @RISK tool.

(i) Uniform functions:

S(U), O(U), D(U): RiskUniform (C4, C3, RiskStatic (9.76))

(ii) Normal functions:

S(N): RiskNormal (K3, K4, RiskStatic (5.17))  
 O(N): RiskNormal (K3, K4, RiskStatic (5.20))  
 D(N): RiskNormal (K3, K4, RiskStatic (5.20))

We define a rounding function in @RISK model to guarantee integer value for all inputs of risk factor SOD. For example, a generated set of risk factors  $(S, O, D) = (1.65, 3.89, 9.26)$  is rounded to  $(2, 4, 9)$ . To ensure that the generated values are in the range of  $[1, 10]$ , we also define a filter in @RISK model. The values smaller than 0.5 or greater than 10.5 are discarded. After 10,000 iterations in the simulation, Table 5 summarizes the statistics of input function and their details.

To validate the proposed model, two stochastic distributions, uniform (U) and normal (N), are simulated up to

TABLE 6: The simulation settings.

|                         |                   |
|-------------------------|-------------------|
| Workbook name           | Simulation models |
| Number of simulations   | 1                 |
| Number of iterations    | 10000             |
| Number of inputs        | 6                 |
| Number of outputs       | 64                |
| Sampling type           | Latin hypercube   |
| Simulation start time   | 5/7/11 19:39:34   |
| Simulation duration     | 00:00:07          |
| Random number generator | Mersenne twister  |
| Random seed             | 404885595         |

10,000 iterations for three risk factors: severity, occurrence, and detectability. There are two parameters, lower bound (LB) and upper bound (UB), defined in the uniform distribution  $U \sim (LB, UB)$ , while mean ( $\mu$ ) and standard deviation ( $\rho$ ) are given in the normal function  $N \sim (\mu, \rho)$ . In this study, simulation settings are defined as  $U \sim (1, 10)$  and  $N \sim (5.5, 1.5)$  for uniform and normal distributions, respectively. The risk weights are assigned to  $L = 0.1, M = 0.3,$  and  $H = 0.6$  throughout the paper. The simulation settings are listed in Table 6.

3.3. Simulation Results

3.3.1. Details of Data Statistics. To easily review the simulation results, we summarize the descriptive statistics of RPN values (TRPN and GRPN) with (1) mean and standard deviation (SD) (mean  $\pm$  SD), (2) skewness, and (3) kurtosis in Table 7, and the distribution sketch is shown in Figure 2.

We describe the central location of the distribution via mean value and the spread via SD. For the GRPN function, the mean values are in a range of 0.67 to 0.73, while the SDs are in a range from 0.05 to 0.18; they are shown as a stable result. For TRPN function, the mean values are around 166, and SDs varied from 56.38 to 153.37.

Skewness is used to measure the asymmetry of the distribution. The GRPN values are all negative in range from  $-0.37$  to  $-0.88$ , while the TRPN values are all positive in range from 0.60 to 1.56. The negative values verify the critique that most TRPN values are not unique, and some are recycled up to 24 times [29]. Kurtosis is used to measure the extent of the distribution peak. For the GRPN function, the kurtosis is in a range from 3.02 to 3.75 and a range from 2.95 to 5.55 in the GRPN function. Applying the RPN-based FMEA model to manage risks, an acceptable risk probability

TABLE 7: Descriptive statistics of mean  $\pm$  SD, skewness, and kurtosis.

| Mean $\pm$ SD,<br>skewness, and<br>kurtosis | Combination of $(w_S, w_O, w_D)$ , GRPN |                                  |                                  |                                  |                                  |                                  |                                  | TRPN                                |
|---------------------------------------------|-----------------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|-------------------------------------|
|                                             | (L, M, H)                               | (L, H, M)                        | (M, L, H)                        | (M, H, L)                        | (H, L, M)                        | (H, M, L)                        | (E, E, E)                        |                                     |
| (S, O, D)                                   |                                         |                                  |                                  |                                  |                                  |                                  |                                  |                                     |
| (UUU)                                       | 0.67 $\pm$ 0.18<br>-0.72<br>3.09        | 0.67 $\pm$ 0.18<br>-0.71<br>3.07 | 0.67 $\pm$ 0.18<br>-0.71<br>3.08 | 0.67 $\pm$ 0.18<br>-0.70<br>3.03 | 0.67 $\pm$ 0.18<br>-0.72<br>3.09 | 0.67 $\pm$ 0.18<br>-0.71<br>3.08 | 0.67 $\pm$ 0.15<br>-0.51<br>3.05 | 165.80 $\pm$ 153.37<br>1.56<br>5.55 |
| (UUN)                                       | 0.67 $\pm$ 0.18<br>-0.72<br>3.09        | 0.69 $\pm$ 0.16<br>-0.85<br>3.08 | 0.71 $\pm$ 0.10<br>-0.61<br>3.18 | 0.68 $\pm$ 0.18<br>-0.72<br>3.03 | 0.69 $\pm$ 0.16<br>-0.85<br>3.10 | 0.68 $\pm$ 0.18<br>-0.73<br>3.07 | 0.69 $\pm$ 0.13<br>-0.60<br>3.03 | 166.30 $\pm$ 124.94<br>1.08<br>3.84 |
| (UNU)                                       | 0.69 $\pm$ 0.16<br>-0.85<br>3.10        | 0.71 $\pm$ 0.10<br>-0.58<br>3.09 | 0.68 $\pm$ 0.18<br>-0.73<br>3.08 | 0.71 $\pm$ 0.10<br>-0.56<br>3.04 | 0.68 $\pm$ 0.18<br>-0.73<br>3.08 | 0.69 $\pm$ 0.16<br>-0.85<br>3.09 | 0.69 $\pm$ 0.13<br>-0.59<br>3.03 | 166.01 $\pm$ 124.85<br>1.11<br>3.96 |
| (UNN)                                       | 0.73 $\pm$ 0.06<br>-0.44<br>3.41        | 0.73 $\pm$ 0.06<br>-0.45<br>3.43 | 0.71 $\pm$ 0.10<br>-0.61<br>3.07 | 0.71 $\pm$ 0.10<br>-0.61<br>3.06 | 0.70 $\pm$ 0.16<br>-0.87<br>3.08 | 0.70 $\pm$ 0.16<br>-0.87<br>3.09 | 0.71 $\pm$ 0.10<br>-0.69<br>3.02 | 166.58 $\pm$ 94.61<br>0.67<br>3.11  |
| (NUU)                                       | 0.68 $\pm$ 0.18<br>-0.74<br>3.09        | 0.68 $\pm$ 0.18<br>-0.73<br>3.07 | 0.69 $\pm$ 0.16<br>-0.85<br>3.11 | 0.69 $\pm$ 0.16<br>-0.85<br>3.08 | 0.71 $\pm$ 0.10<br>-0.59<br>3.14 | 0.71 $\pm$ 0.10<br>-0.59<br>3.10 | 0.69 $\pm$ 0.13<br>-0.59<br>3.04 | 166.40 $\pm$ 125.19<br>1.08<br>3.79 |
| (NUN)                                       | 0.71 $\pm$ 0.10<br>-0.66<br>3.25        | 0.70 $\pm$ 0.16<br>-0.88<br>3.10 | 0.73 $\pm$ 0.06<br>-0.48<br>3.50 | 0.70 $\pm$ 0.16<br>-0.88<br>3.09 | 0.73 $\pm$ 0.06<br>-0.50<br>3.59 | 0.71 $\pm$ 0.10<br>-0.66<br>3.16 | 0.71 $\pm$ 0.10<br>-0.73<br>3.10 | 166.56 $\pm$ 93.80<br>0.60<br>2.95  |
| (NNU)                                       | 0.70 $\pm$ 0.16<br>-0.88<br>3.11        | 0.71 $\pm$ 0.10<br>-0.63<br>3.17 | 0.70 $\pm$ 0.16<br>-0.88<br>3.12 | 0.73 $\pm$ 0.06<br>-0.47<br>3.49 | 0.71 $\pm$ 0.10<br>-0.64<br>3.21 | 0.73 $\pm$ 0.06<br>-0.47<br>3.58 | 0.71 $\pm$ 0.10<br>-0.72<br>3.13 | 166.40 $\pm$ 94.14<br>0.65<br>3.03  |
| (NNN)                                       | 0.73 $\pm$ 0.06<br>-0.51<br>3.57        | 0.73 $\pm$ 0.06<br>-0.54<br>3.71 | 0.73 $\pm$ 0.06<br>-0.49<br>3.48 | 0.73 $\pm$ 0.06<br>-0.53<br>3.67 | 0.73 $\pm$ 0.06<br>-0.52<br>3.75 | 0.73 $\pm$ 0.06<br>-0.53<br>3.75 | 0.73 $\pm$ 0.05<br>-0.39<br>3.30 | 166.68 $\pm$ 56.38<br>0.64<br>3.59  |

TABLE 8: Threshold value ( $T_i$ ) for each of the combinations with  $\alpha = 0.9$ .

| $T_i$     | Combination of $(w_S, w_O, w_D)$ , GRPN |           |           |           |           |           |           | TRPN |
|-----------|-----------------------------------------|-----------|-----------|-----------|-----------|-----------|-----------|------|
|           | (L, M, H)                               | (L, H, M) | (M, L, H) | (M, H, L) | (H, L, M) | (H, M, L) | (E, E, E) |      |
| (S, O, D) |                                         |           |           |           |           |           |           |      |
| (UUU)     | 0.88                                    | 0.88      | 0.88      | 0.88      | 0.88      | 0.88      | 0.86      | 378  |
| (UUN)     | 0.88                                    | 0.87      | 0.83      | 0.89      | 0.87      | 0.89      | 0.85      | 350  |
| (UNU)     | 0.87                                    | 0.83      | 0.89      | 0.83      | 0.89      | 0.87      | 0.85      | 350  |
| (UNN)     | 0.80                                    | 0.80      | 0.83      | 0.83      | 0.87      | 0.87      | 0.82      | 294  |
| (NUU)     | 0.89                                    | 0.89      | 0.87      | 0.87      | 0.83      | 0.83      | 0.85      | 350  |
| (NUN)     | 0.83                                    | 0.87      | 0.80      | 0.87      | 0.80      | 0.83      | 0.82      | 294  |
| (NNU)     | 0.87                                    | 0.83      | 0.87      | 0.80      | 0.83      | 0.80      | 0.82      | 294  |
| (NNN)     | 0.80                                    | 0.80      | 0.80      | 0.80      | 0.80      | 0.80      | 0.80      | 245  |

$\alpha$  must be defined, which depends on the risk preference of organization, department, or process. Given  $\alpha$ , a threshold can be precalculated where  $\text{Prob}(\text{GRPN} \leq T_i) = \alpha$ , as shown in Figure 1. Whenever the GRPN value  $> T_i$ , priority should be given to taking corrective action against the failure modes. In this paper, we assign  $\alpha = 0.9$  as an example; then threshold values ( $T_i$ ) with respective risk factors and

the risk weights for each of combinations are suggested in Table 8. The threshold values for GRPN function are in the range from 0.80 to 0.89, while the threshold values for TRPN function vary from 245 to 378. Potential failure modes whose value are greater than the threshold in respective scenarios must be taken corrective actions. If several failure modes are more critical, we can assign  $\alpha$  for them with a smaller value.

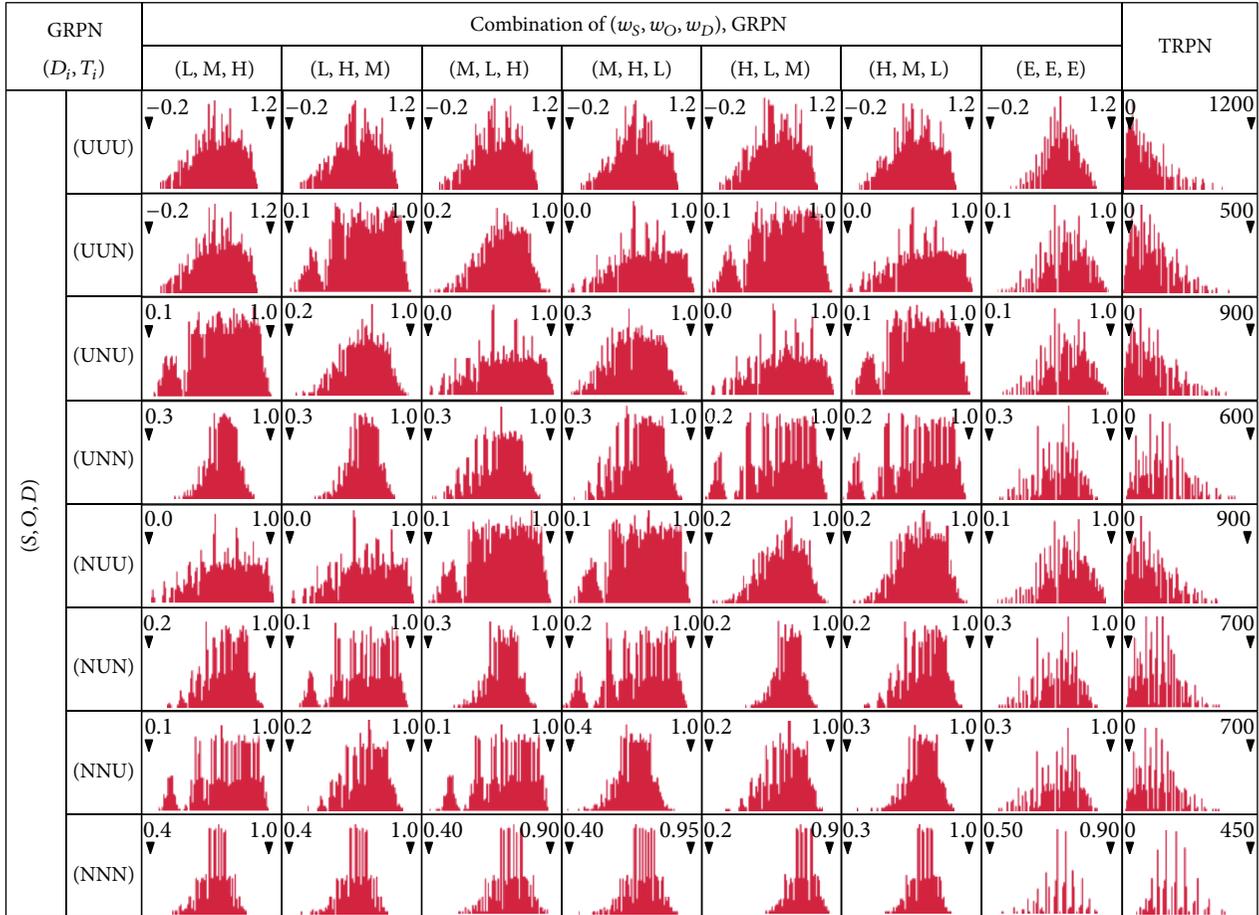


FIGURE 2: Comparison of correlation coefficients for the risk factors.

The smaller the acceptable risk probability ( $\alpha$ ), the larger the possibility that the GRPN values that are contributed by the failure modes will be greater than the  $T_i$ .

### 4. Sensitivity Analysis

To evaluate the function’s stability, we perform sensitivity analysis on both correlation and regression. They are generated from the @RISK built-in function.

**4.1. Correlation Coefficient.** To validate that the proposed GRPN-based model dominates the TRPN-based model, the GRPN function is equivalent to TRPN when the risk weight is assigned with (EEE). Moreover, we compare the correlation coefficients of the risk factors (SOD) with different distributions (uniform and normal) respective to function values, that is, GRPN with weight (EEE) and TRPN. For the risk factor  $S$  in Figure 3, the values of both functions are almost the same, except the (UNN) distribution. They are 0.871 and 0.612 for EEE and TRPN, respectively. Regarding the risk factor  $O$ , both the functions are similar because the lines between each function overlap. Again, both functions are almost the same for the risk factor  $D$  except for the distribution (UUN). They are 0.348 and 0.203 for EEE and

TRPN, respectively. According to the correlation coefficient, both the functions are highly correlated. This implies that the GRPN function is equivalent to TRPN function when the weight (EEE) is assigned.

**4.2. Regression Coefficient.** Regression analysis is used to investigate the relationship between the risk factors (independent variables) and RPN function value (dependent variable, that is, GRPN/TRPN function value). The coefficient of determination  $R^2$  is used in the context of statistical models. The primary objective is to predict future outcomes on the basis of other related information. It is the proportion of variability in a dataset that is accounted for the statistical model. It also provides a measure of how well the future outcomes are likely to be predicted by the model.

In Figure 4, we illustrate the  $R^2$  values for all combinations. Regardless of the risk weight (LMH, LHM, MLH, MHL, HLM, HML, or EEE) assigned, the GRPN function has stable  $R^2$  values about 0.9; they are in a range from 0.894 to 0.906. The results show that the proposed GRPN function outperforms TRPN function because the  $R^2$  of the GRPN function is stable and greater than that of the TRPN function. An interesting finding is that the  $R^2$  values of TRPN

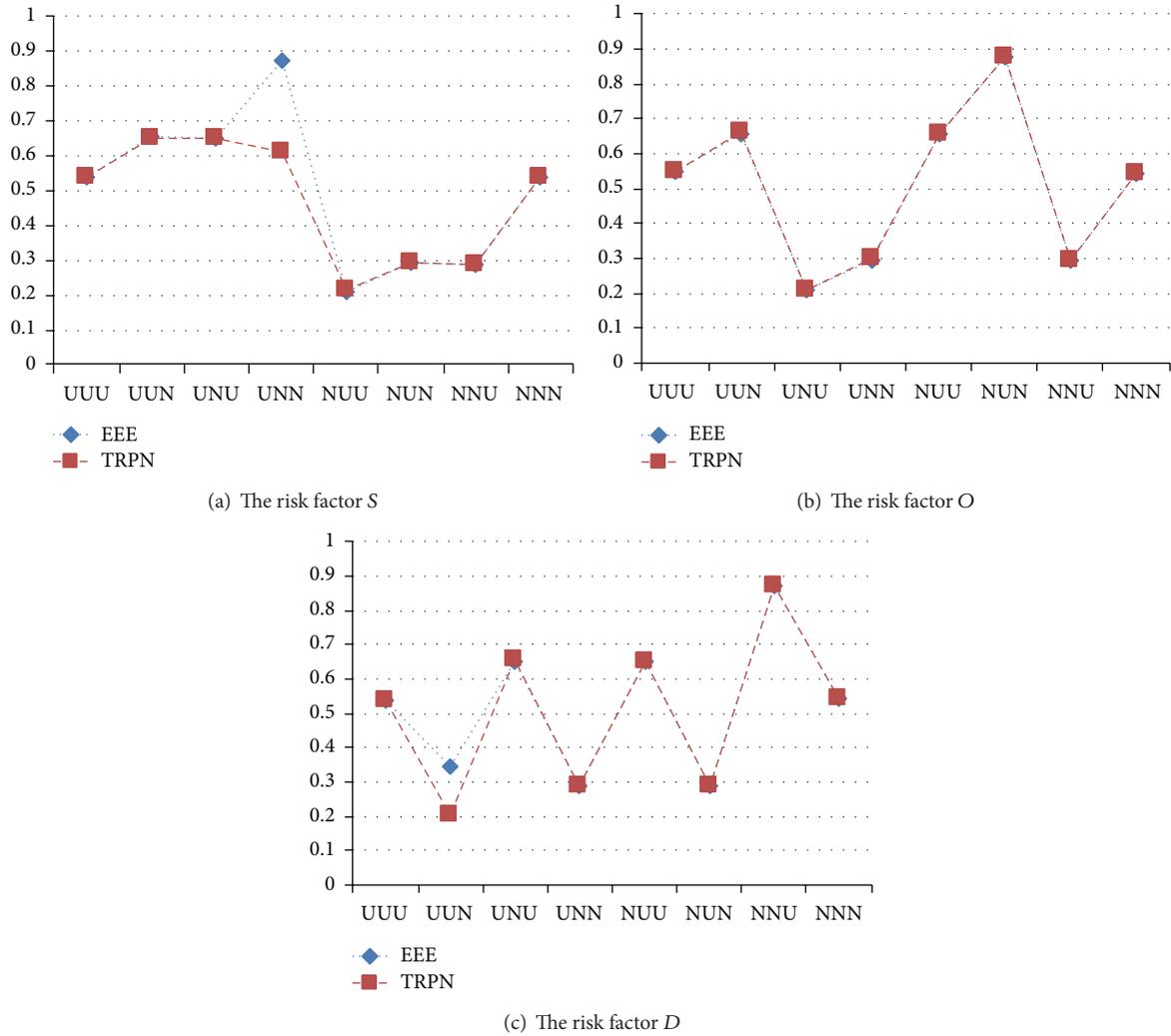


FIGURE 3: Distribution sketch with respect to risk factors and risk weights.

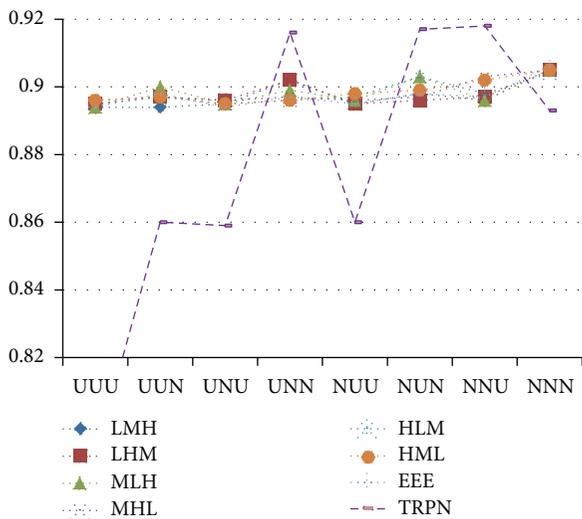


FIGURE 4:  $R^2$  comparison of eight distributions with respect to several risk weights.

function are larger than that of GRPN function in three special cases of SOD distribution, that is, UNN, NUN, and NNU. The TRPN function is suitable for cases in which the majority of the three risk factors are in normal distribution, whereas the GRPN function is suitable for the others. In general, the proposed GRPN function offers a more adaptive approach, which can be applied in industries with various risk preferences.

### 5. Case Study

5.1. An Example of Down Syndrome Test. We use an example of Down syndrome test—a healthcare application—to explain the operation of the proposed GRPN model. It is to identify significant achievement and its adaptability compared with that of the TRPN model. In addition, we consider three scenarios to demonstrate the adaptability of the proposed model. The steps are as follows:

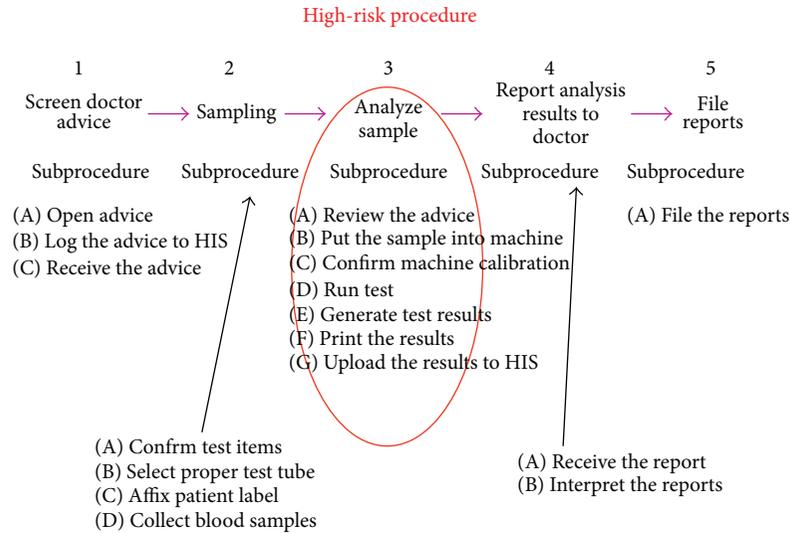


FIGURE 5: The test procedure for Down syndrome.

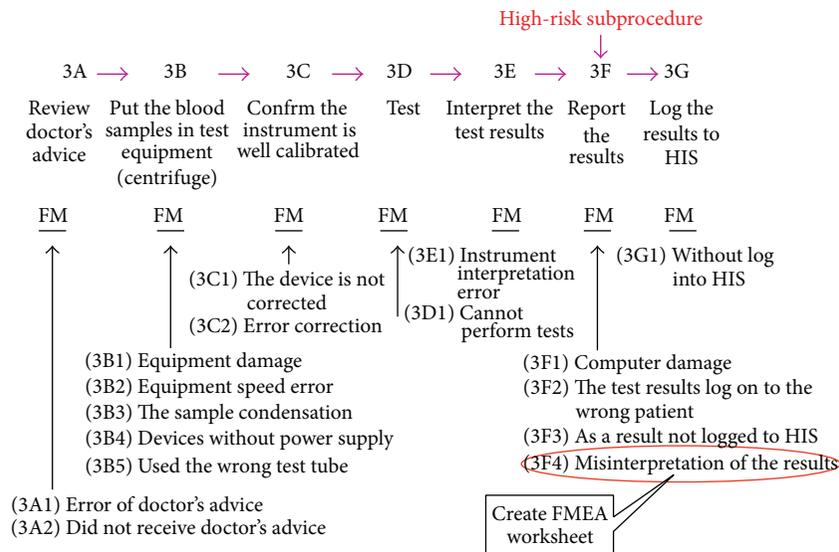


FIGURE 6: The high-risk subprocedures and the failure modes in Step 3.

Step 1. Select a procedure for study, as shown in Figure 5.

Step 2. Assemble a team to monitor the failure mode.

Step 3. Compile an operational risk analysis flowchart. If the third stage (analysis of the sample) involves a high-risk procedure, it is also necessary to implement the following (sub)procedures: (3A) review the doctor's advice; (3B) put the blood sample into the machine for testing; (3C) confirm that the machine is calibrated; (3D) run the test; (3E) generate the test results; (3F) print the results; and (3G) upload the results to the health information system (HIS). The results of the high-risk subprocedures (step 3F) and their failure modes are shown in Figure 6. The potential failure

modes are damage to the computer, the test results are logged to the wrong patient's file, the results are not uploaded to the HIS, and misinterpretation of the results.

Step 4. Identify the potential failure modes via a group discussion. In Figure 6, the misinterpretation of the results (step 3F4) is probably a failure mode (FM).

Step (4-1). Identify the reasons for the failure, and determine its severity, occurrence, and detectability. For each failure mode, the root cause(s) of the failure should be determined. The failure "misinterpretation of the results" has four possible causes, namely, "too tired," "too busy,"

TABLE 9: Step 3F4 worksheet: misinterpretation of the results.

| Reasons for failure | Step 4: failure risk analysis Rating* |   |      | Decision tree          |                     |           | Step 5: correction/action/result |                                                              |                                                |                        |
|---------------------|---------------------------------------|---|------|------------------------|---------------------|-----------|----------------------------------|--------------------------------------------------------------|------------------------------------------------|------------------------|
|                     | S                                     | D | GRPN | With Control mechanism | Problems Detectable | Continued | Correction                       | Action                                                       | Result                                         | Role in charge         |
| 3F4a                | 9                                     | 8 | 2    | 0.874                  | No                  | Yes       | Control                          | Confirmed by second examination                              | Confirmed by two examiners                     | Laboratory director    |
| 3F4b                | 3                                     | 9 | 8    | 0.663                  | No                  | No        | Control                          | Personnel access control and provision of a single telephone | Replacement of laboratory and telephone system | General department     |
| 3F4c                | 2                                     | 8 | 8    | 0.542                  | No                  | No        | Provide a second light source    | Installation of new lights                                   | Brighter lights                                | General department     |
| 3F4d                | 8                                     | 6 | 4    | 0.836                  | No                  | Yes       | Need to deal with                | Purchase new equipment                                       | Equipment installed on a given date (YY/MM/DD) | Procurement department |

\*  $w_S = 0.6$ ,  $w_D = 0.3$ , and  $w_D = 0.1$ .

TABLE 10: Comparison of the TRPN and GRPN test procedures—scenario 1.

| FM   | Risk factor |   |   | TRPN       |                                            |               | GRPN         |                                            |               |
|------|-------------|---|---|------------|--------------------------------------------|---------------|--------------|--------------------------------------------|---------------|
|      | S           | O | D | Value      | Mean value of the acceptable threshold $T$ | Identify risk | Value        | Mean value of the acceptable threshold $T$ | Identify risk |
| 3F4a | 9           | 8 | 2 | <i>144</i> | 166                                        | N             | <i>0.876</i> | 0.74                                       | Y             |
| 3F4b | 3           | 9 | 8 | <i>216</i> | 166                                        | Y             | <i>0.663</i> | 0.74                                       | N             |
| 3F4c | 2           | 8 | 8 | 128        | 166                                        | N             | 0.542        | 0.74                                       | N             |
| 3F4d | 8           | 6 | 4 | <i>192</i> | 166                                        | Y             | <i>0.836</i> | 0.74                                       | Y             |

TABLE 11: Comparison of the TRPN and GRPN test procedures—scenario 2.

| FM   | Risk factor |   |   | TRPN       |                                            |               | GRPN         |                                            |               |
|------|-------------|---|---|------------|--------------------------------------------|---------------|--------------|--------------------------------------------|---------------|
|      | S           | O | D | Value      | Mean value of the acceptable threshold $T$ | Identify risk | Value        | Mean value of the acceptable threshold $T$ | Identify risk |
| 3F4a | 9           | 8 | 2 | 144        | 166                                        | N             | 0.557        | 0.74                                       | N             |
| 3F4b | 3           | 9 | 8 | 216        | 166                                        | Y             | 0.780        | 0.74                                       | Y             |
| 3F4c | 2           | 8 | 8 | 128        | 166                                        | N             | 0.722        | 0.74                                       | N             |
| 3F4d | 8           | 6 | 4 | <i>192</i> | 166                                        | Y             | <i>0.710</i> | 0.74                                       | N             |

“insufficient light,” and “misunderstanding of the machine report.”

Step (4-2). Compile an FMEA worksheet (as shown in Table 9).

Step (4-3). Sort the failure modes by their GRPN values.

Step 5. Take corrective action if the GRPN value of a failure mode is higher than a given threshold.

5.2. Comparison of the GRPN and RPN Test Procedures. To differentiate between GRPN-based FMEA model and the TRPN-based FMEA approach, we consider different values for the weights ( $w_S, w_O, w_D$ ). Let \*\*\* denote weight “H” (value 0.6), \*\* denote weight “M” (value 0.3), and \* denote weight “L” (value 0.1). In addition, we set the threshold value for traditional RPN functions at 166 (it is the average RPN value = SOD) [20] and calculate an equivalent threshold value (with SOD = 166) for the GRPN at 0.74. We also give an average value of the weights (equal weight),  $w_S = w_O = w_D = 0.333333$ ; then the GRPN value is  $\log(S^{w_S} \cdot O^{w_O} \cdot D^{w_D}) = w_S \log S + w_O \log O + w_D \log D = w_S * (\log S + \log O + \log D) = w_S * \log(\text{SOD}) = 0.333333 * \log(166) = 0.74$ . With an equivalent value, the following scenarios demonstrate adaptability of the proposed GRPN approach. Irrespective of the approach applied, corrective action should be taken on the FMs whose values are greater than the given thresholds. We compare three scenarios of ( $w_S, w_O, w_D$ ): (H, M, L), (M, L, H), and (L, H, M).

5.2.1. Scenario 1 ( $w_S^{***}, w_O^{**}, w_D^*$ ). This scenario focuses on the factor S. It is assumed that the preferences for the SOD weights are  $w_S = 0.6, w_O = 0.3, \text{ and } w_D = 0.1$ , as shown in Table 10. The italic indicates the differentiation of risk identification from the GRPN function to the TRPN function. The values are  $144(9 \times 8 \times 2), 216(3 \times 9 \times 8),$

$128(2 \times 8 \times 8),$  and  $192(8 \times 6 \times 4),$  and the GRPN values are  $0.876 = \log(9^{0.6} \times 8^{0.3} \times 2^{0.1}), 0.663 = \log(3^{0.6} \times 9^{0.3} \times 8^{0.1}), 0.542 = \log(2^{0.6} \times 8^{0.3} \times 8^{0.1}),$  and  $0.836 = \log(8^{0.6} \times 6^{0.3} \times 4^{0.1}),$  for 3F4a, 3F4b, 3F4c, and 3F4d, respectively. By using the proposed GRPN model, we can identify the risk of failure mode (FM) 3F4a, but the FM is ignored (no corrective actions will be taken on the FM) by the traditional RPN approach. For FM 3F4b, the GRPN approach ignores the failure (without corrective actions); however, the traditional RPN approach identifies the FM.

5.2.2. Scenario 2 ( $w_S^{**}, w_O^*, w_D^{***}$ ). This scenario focuses on the factor D. It is assumed that the preferences for the SOD weights are  $w_S = 0.3, w_O = 0.1, \text{ and } w_D = 0.6,$  as shown in Table 11. The TRPN values are  $144(9 \times 8 \times 2), 216(3 \times 9 \times 8), 128(2 \times 8 \times 8),$  and  $192(8 \times 6 \times 4),$  and the GRPN values are  $0.557 = \log(9^{0.3} \times 8^{0.1} \times 2^{0.6}), 0.780 = \log(3^{0.3} \times 9^{0.1} \times 8^{0.6}), 0.722 = \log(2^{0.3} \times 8^{0.1} \times 8^{0.6}),$  and  $0.710 = \log(8^{0.3} \times 6^{0.1} \times 4^{0.6}),$  for 3F4a, 3F4b, 3F4c, and 3F4d, respectively. For FM 3F4d, the GRPN approach ignores the FM, but the traditional RPN approach identifies it.

5.2.3. Scenario 3 ( $w_S^*, w_O^{***}, w_D^{**}$ ). This scenario focuses on the factor O. It is assumed that the preferences for the SOD weights are  $w_S = 0.1, w_O = 0.6, \text{ and } w_D = 0.3,$  as shown in Table 12. The TRPN values are  $144(9 \times 8 \times 2), 216(3 \times 9 \times 8), 128(2 \times 8 \times 8),$  and  $192(8 \times 6 \times 4),$  and the GRPN values are  $0.728 = \log(9^{0.1} \times 8^{0.6} \times 2^{0.3}), 0.891 = \log(3^{0.1} \times 9^{0.6} \times 8^{0.3}), 0.843 = \log(2^{0.1} \times 8^{0.6} \times 8^{0.3}),$  and  $0.738 = \log(8^{0.1} \times 6^{0.6} \times 4^{0.3}),$  for 3F4a, 3F4b, 3F4c, and 3F4d, respectively. By using the proposed GRPN model, we can identify a risk in the FM 3F4c, but the traditional RPN approach ignores the risk. GRPN ignores the FM 3F4d; however, the traditional RPN approach can identify the FM.

TABLE 12: Comparison of the TRPN and GRPN test procedures—scenario 3.

| FM   | Risk factor |   |   | TRPN  |                                            |               | GRPN  |                                            |               |
|------|-------------|---|---|-------|--------------------------------------------|---------------|-------|--------------------------------------------|---------------|
|      | S           | O | D | Value | Mean value of the acceptable threshold $T$ | Identify risk | Value | Mean value of the acceptable threshold $T$ | Identify risk |
| 3F4a | 9           | 8 | 2 | 144   | 166                                        | N             | 0.728 | 0.74                                       | N             |
| 3F4b | 3           | 9 | 8 | 216   | 166                                        | Y             | 0.891 | 0.74                                       | Y             |
| 3F4c | 2           | 8 | 8 | 128   | 166                                        | N             | 0.843 | 0.74                                       | Y             |
| 3F4d | 8           | 6 | 4 | 192   | 166                                        | Y             | 0.738 | 0.74                                       | N             |

## 6. Application on E-Healthcare

With the development of information technology, in recent years, it will be an increased focus on healthcare that is user-centered in design in an attempt to meet demand. It also is one of the fastest growing areas of healthcare provision [31]. An integrated framework of e-healthcare service is proposed and it consists of both architecture design and network transmission design [32]. The e-healthcare equipment is used as a tool in the management of long-term conditions in the community to proactively monitor patients and respond promptly to indicators of acute exacerbations. For example, care receivers are trained to operate a device which measures physiological indices such as blood pressure, oxygen saturations and pulse, spirometry, temperature, ECG, and blood glucose readings each day in their home. All devices can be individually programmed to suit the lifestyle and day to day living habits of the person. Generally speaking, the caregivers/care providers take most of the decision-making responsibility and play an important role in healthcare environment which is human intensive task and intention-aware systems that outperform situation-aware systems can eliminate unnecessary humans involved [33]. With the constantly growing information in ubiquitous environment, for example, Internet of Things (IoT), quality and reliability of healthcare sensors has become the new strategic challenge for care providers that aim to capture the whole healthcare information. A data mining-based knowledge mapping approach is proposed to improve the process of acquiring knowledge for healthcare [34]. Even e-healthcare is a convenient approach for improving care access for the care receivers; one of three criteria to evaluate the effectiveness is quality of e-healthcare service [35]. An example of e-healthcare architecture is shown in Figure 7, in which three levels of services can be organized:

- (i) infrastructure level [14, 17, 32, 36–39]: endpoint device (vital sign sensor, POC detector), data transmission (Bluetooth, Zegbee, Wi-Fi, 3G+, Internet), middleware (Gateway, data exchange, HL7, LOINC, etc.), care system (call center, e-healthcare IS, HIS, etc.);
- (ii) system level [17, 32, 36]: user interface, data processing, data exchange, data repository.
- (iii) Data source level [32, 37, 40]: sensor data, HIS, disease IS, clinical interview.

Due to the complexity of e-healthcare service environment, we present a generic modeling of failure risk analysis

for the environment, as shown in Figure 8. Define  $L$  as the number of levels in e-healthcare service hierarchy,  $S_l$  ( $l \in L$ ) as the number of service sets in  $l$ -level and  $S_{ls}$  as  $s$ -set in  $l$ -level, and  $E_{ls}$  ( $l \in L, s \in S_l$ ) as the number of service elements in the  $s$ -set of  $l$ -level. Then, we define  $R_{lse}$  as the  $e$ -element ( $e \in E_{ls}$ ) of e-healthcare service, where  $R_{lse}$  ( $e \in E_{ls}$ ) belongs to the service set  $S_{ls}$ .

*Definition 1.* Risk( $S$ ) is the risk of the entire e-healthcare service  $S$ , where Risk( $S$ ) = Risk( $S_{11}$ ) and is a function of  $R_{lse}$  ( $l = 1, s = 1, e \in E_{1s}$ ), because  $S_{11}$  is the first/highest level of the  $S$  and the only one service set in the first level.

*Definition 2.* Risk( $S_{ls}$ ), where  $l > 1$ , is the risk of the service set  $s$  in the level  $l$ . Moreover, the risk  $R_{lse}$  is derived from the service set in lower level  $l + 1$ . For example, the set  $S_{ls} = \{R_{ls1}, R_{ls2}, \dots, R_{lsE_{ls}}\}$ , each of service elements,  $R_{lse}$ , is recursively expanded to the respective service sets in next level and there are  $S_{l+1,k}, S_{l+1,k+1}, \dots, S_{l+1,k+E_{ls}}$ .

*Definition 3.* GRPN( $R_{lse}$ ) is the value of GRPN function defined in (2). For each  $R_{lse}$  in  $S_{ls}$ , the value can be expressed as (5), where  $w_{S_{lse}}, w_{O_{lse}}$ , and  $w_{D_{lse}}$  are the weights given for the service element  $e$  in the service set  $s$  of the level  $l$ . Consider

$$\text{GRPN}(R_{lse}) = w_{S_{lse}} \log S_{lse} + w_{O_{lse}} \log O_{lse} + w_{D_{lse}} \log D_{lse}. \quad (5)$$

*Property 1* (risk analysis/identification for the entire service  $S$ ). Risk( $S$ ) =  $\{R_{lse} \mid \text{GRPN}(R_{lse}) \geq T_{1s}, l = 1, s = 1, e \in E_{1s}\}$ , where  $T_{1s}$  is the acceptable level of the risk value (GRPN threshold) for the e-healthcare service, as defined in Section 3.1.2.

*Property 2* (risk analysis/identification for the service  $S_{ls}$ ). Risk( $S_{ls}$ ) =  $\{R_{lse} \mid \text{GRPN}(R_{lse}) \geq T_{ls}, l > 1, e \in E_{ls}\}$ , where  $T_{ls}$  ( $l \in L, s \in S_l$ ) is the acceptable level of the risk value (GRPN threshold) for the e-healthcare service.

The acceptable level of risk value,  $T_{ls}$ , can be all the same or different according to the requirement of risk management policy. Based upon the properties 1 and 2, we can identify the potential risks of e-healthcare service. To illustrate the capability of the proposed adaptable risk identification model, we present a simple example to differentiate the risk of service elements with the hierarchical architecture of e-healthcare environment. Referring to Figure 7, if there are three elements in the infrastructure level: endpoint device ( $R_{111}$ ), data transmission ( $R_{112}$ ), and care system ( $R_{113}$ );

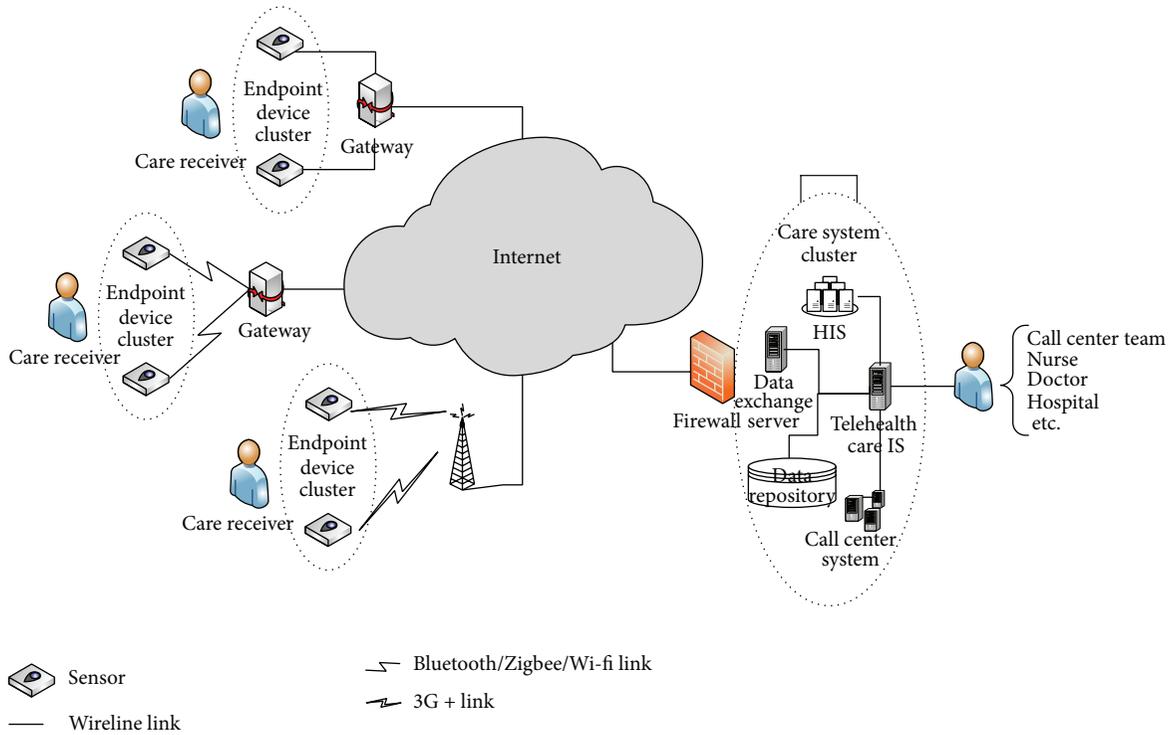


FIGURE 7: An example of e-healthcare architecture.

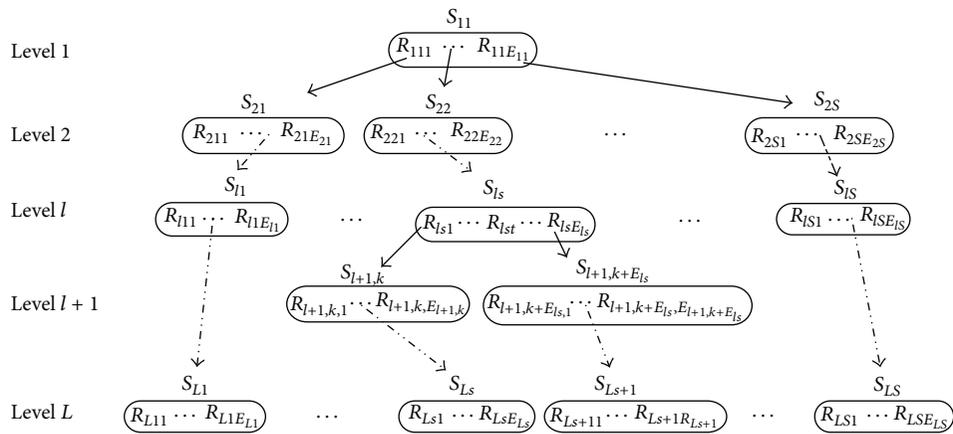


FIGURE 8: Hierarchy of e-healthcare service architecture.

TABLE 13: Adaptive risk identification with scenario of risk weight threshold: homo-homo.

| Elements in level 1 | Risk factor |     |     | Risk weight |       |       | Risk ( $S_{ls}$ )  |          |               |
|---------------------|-------------|-----|-----|-------------|-------|-------|--------------------|----------|---------------|
|                     | $S$         | $O$ | $D$ | $w_S$       | $w_O$ | $w_D$ | GRPN ( $R_{lse}$ ) | $T_{ls}$ | Identify risk |
| $R_{111}$           | 5           | 8   | 1   | 0.6         | 0.3   | 0.1   | 0.69               | 0.74     | No            |
| $R_{112}$           | 5           | 8   | 4   | 0.6         | 0.3   | 0.1   | 0.75               | 0.74     | Yes           |
| $R_{113}$           | 2           | 5   | 8   | 0.6         | 0.3   | 0.1   | 0.48               | 0.74     | No            |
| $R_{114}$           | 2           | 9   | 3   | 0.6         | 0.3   | 0.1   | 0.51               | 0.74     | No            |
| $R_{115}$           | 9           | 4   | 2   | 0.6         | 0.3   | 0.1   | 0.78               | 0.74     | Yes           |
| $R_{116}$           | 5           | 9   | 5   | 0.6         | 0.3   | 0.1   | 0.78               | 0.74     | Yes           |
| $R_{117}$           | 5           | 2   | 4   | 0.6         | 0.3   | 0.1   | 0.57               | 0.74     | No            |

TABLE 14: Adaptive risk identification with scenario of risk weight threshold: homo-hetero.

| Elements in level 1 | Risk factor |   |   | Risk weight |       |       | Risk ( $S_{ls}$ )  |          | Identify risk |
|---------------------|-------------|---|---|-------------|-------|-------|--------------------|----------|---------------|
|                     | S           | O | D | $w_S$       | $w_O$ | $w_D$ | GRPN ( $R_{lse}$ ) | $T_{ls}$ |               |
| $R_{111}$           | 5           | 8 | 1 | 0.6         | 0.3   | 0.1   | 0.69               | 0.55     | Yes           |
| $R_{112}$           | 5           | 8 | 4 | 0.6         | 0.3   | 0.1   | 0.75               | 0.78     | No            |
| $R_{113}$           | 2           | 5 | 8 | 0.6         | 0.3   | 0.1   | 0.48               | 0.50     | No            |
| $R_{114}$           | 2           | 9 | 3 | 0.6         | 0.3   | 0.1   | 0.51               | 0.62     | No            |
| $R_{115}$           | 9           | 4 | 2 | 0.6         | 0.3   | 0.1   | 0.78               | 0.75     | Yes           |
| $R_{116}$           | 5           | 9 | 5 | 0.6         | 0.3   | 0.1   | 0.78               | 0.61     | Yes           |
| $R_{117}$           | 5           | 2 | 4 | 0.6         | 0.3   | 0.1   | 0.57               | 0.55     | Yes           |

TABLE 15: Adaptive risk identification with scenario of risk weight threshold: hetero-homo.

| Elements in level 1 | Risk factor |   |   | Risk weight |       |       | Risk ( $S_{ls}$ )  |          | Identify risk |
|---------------------|-------------|---|---|-------------|-------|-------|--------------------|----------|---------------|
|                     | S           | O | D | $w_S$       | $w_O$ | $w_D$ | GRPN ( $R_{lse}$ ) | $T_{ls}$ |               |
| $R_{111}$           | 5           | 8 | 1 | 0.6         | 0.3   | 0.1   | 0.69               | 0.74     | No            |
| $R_{112}$           | 5           | 8 | 4 | 0.3         | 0.6   | 0.1   | 0.81               | 0.74     | Yes           |
| $R_{113}$           | 2           | 5 | 8 | 0.3         | 0.1   | 0.6   | 0.70               | 0.74     | No            |
| $R_{114}$           | 2           | 9 | 3 | 0.6         | 0.1   | 0.3   | 0.42               | 0.74     | No            |
| $R_{115}$           | 9           | 4 | 2 | 0.6         | 0.1   | 0.3   | 0.72               | 0.74     | No            |
| $R_{116}$           | 5           | 9 | 5 | 0.3         | 0.6   | 0.1   | 0.85               | 0.74     | Yes           |
| $R_{117}$           | 5           | 2 | 4 | 0.6         | 0.3   | 0.1   | 0.57               | 0.74     | No            |

three elements in the system level: user interface ( $R_{114}$ ), data processing ( $R_{115}$ ), and data exchange ( $R_{116}$ ); one element in the data source level: sensor data ( $R_{117}$ ), each of elements can be further recursively divided into respective services ( $S_{ls}$ ,  $l > 1$ ) in higher levels, in which potential risks are to be identified.

In this example, we only focus on seven elements of service  $S_{11}$  in level 1; they are  $R_{111}$ ,  $R_{112}$ ,  $R_{113}$ ,  $R_{114}$ ,  $R_{115}$ ,  $R_{116}$ , and  $R_{117}$ . Moreover, model adaptability is shown with parameter combination of both risk weight ( $w_{S_{lse}}$ ,  $w_{O_{lse}}$ ,  $w_{D_{lse}}$ ) and acceptable risk threshold ( $T_{ls}$ ). Each of them is further separated into homogeneous (homo) and heterogeneous (hetero) cases between seven elements ( $R_{111} \sim R_{117}$ ). The case homo means values assigned to the parameter are all the same, while the case hetero means values assigned to the parameter are different. Accordingly, there are four scenarios of risk weight-threshold combination: homo-homo, home-hetero, hetero-home, and hetero-hetero; they are illustrated in Tables 13, 14, 15, and 16, respectively. From the results of four scenarios analysis, only two elements ( $R_{114}$  and  $R_{116}$ ) get identical suggestion, without risk identification for  $R_{114}$  and with risk identification for  $R_{116}$ . The proposed adaptive approach is capable of differentiating the other five elements with regard to different risk preferences.

## 7. Conclusion

FMEA has long been used to evaluate the safety and reliability of products and services in a number of industries. The traditional FMEA model uses the RPN number to prioritize failure modes. Since the three indices used to calculate the RPN are ordinal scale variables, the product of the three ordinal

numbers cannot reflect the actual costs incurred by failures. As a result, the traditional model cannot provide precise information about failure risks, such as the probabilities of the severity, occurrence, and detectability factors. In addition, it is difficult to apply the traditional FMEA to various risk preferences. To overcome these limitations, we propose a generic RPN model called GRPN-based FMEA, which allows us to evaluate the risk factors and their relative weights in a linear manner rather than in a nonlinear relationship. The model uses the logarithm function to assess the severity, occurrence, and detectability factors. It also represents the risk value (GRPN) as a risk factor and a risk weight in a linear relationship, instead of the nonlinear approach used in the traditional RPN formulation. The result shows that the proposed model outperforms the TRPN model. The proposed model provides a practical, effective, and adaptive method for risk evaluation in FMEA. In particular, the defined GRPN offers a new way to prioritize failure modes in FMEA. The different risk preferences considered in the healthcare example show that the modified FMEA model can take account of the various risk factors and prioritize failure modes more accurately. Moreover, with the constantly increasing requirement of e-healthcare service, we also propose a generic modeling of failure risk analysis for the service. The model is capable of adaptively identifying the failure risks in a hierarchical service architecture.

This paper proposes a generic RPN (GRPN) function-based FMEA model for risk analysis that assigns a weight (risk weight) to each risk factor so that the weights represent individual organization/department/process preferences for the factors. To validate the proposed model, the risk factors are randomly generated with both uniform and normal distributions via a simulation process. We also conduct sensitivity

TABLE 16: Adaptive risk identification with scenario of risk weight threshold: hetero-hetero.

| Elements in level 1 | Risk factor |   |   | Risk weight |       |       | Risk ( $S_{Is}$ )  |          | Identify risk |
|---------------------|-------------|---|---|-------------|-------|-------|--------------------|----------|---------------|
|                     | S           | O | D | $w_S$       | $w_O$ | $w_D$ | GRPN ( $R_{lse}$ ) | $T_{Is}$ |               |
| $R_{111}$           | 5           | 8 | 1 | 0.6         | 0.3   | 0.1   | 0.69               | 0.55     | Yes           |
| $R_{112}$           | 5           | 8 | 4 | 0.3         | 0.6   | 0.1   | 0.81               | 0.78     | Yes           |
| $R_{113}$           | 2           | 5 | 8 | 0.3         | 0.1   | 0.6   | 0.70               | 0.50     | Yes           |
| $R_{114}$           | 2           | 9 | 3 | 0.6         | 0.1   | 0.3   | 0.42               | 0.62     | No            |
| $R_{115}$           | 9           | 4 | 2 | 0.6         | 0.1   | 0.3   | 0.72               | 0.75     | No            |
| $R_{116}$           | 5           | 9 | 5 | 0.3         | 0.6   | 0.1   | 0.85               | 0.61     | Yes           |
| $R_{117}$           | 5           | 2 | 4 | 0.6         | 0.3   | 0.1   | 0.57               | 0.55     | Yes           |

analysis on correlation and regression to compare it to the traditional (TRPN-based) approach. To understand how the proposed model works, we use a healthcare example as a potential application of the proposed GRPN-based FMEA model. An illustrated example of Down syndrome test is given, and the computation of GRPNs is explained in detail.

We introduce two application modes based on experience and preference. The experience-based mode allows the user to choose a risk factor combination arbitrarily. This mode can be used in different organizations, departments, or processes, by estimating historical data of failure modes for each of the risk factors (SOD). Under the preference-based mode, we assume that the organization always defines a risk management policy to identify failure modes. Therefore, the weight combination is determined by the policy, for example, (H, L, M). After selecting the weight combination, we set the GRPN threshold to determine if the failure modes exist. However, this paper only discusses two of various stochastic models for the risk factor distribution, that is, uniform and normal. In fact, there are numerous distributions in real world. More realistically, future work can pay more attention to testing and validation for various distributions.

## Conflict of Interests

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

## Acknowledgment

This paper is sponsored by the National Science Council of Taiwan (NSC 98-2410-H-227-004).

## References

- [1] R. Y. Shtykh and Q. Jin, "A human-centric integrated approach to web information search and sharing," *Human-Centric Computing and Information Sciences*, vol. 1, p. 2, 2011.
- [2] N. Y. Yen and S. Y. F. Kuo, "An intergrated approach for internet resources mining and searching," *Journal of Convergence*, vol. 3, no. 2, pp. 37–44, 2012.
- [3] D. Hubbard, *The Failure of Risk Management: Why It's Broken and How to Fix It*, John Wiley & Sons, New York, NY, USA, 2009.
- [4] ISO, *ISO/IEC Guide 73:2009 Risk Management-Vocabulary*, 2009.
- [5] H. VanCott, "Human errors: their causes and reduction," in *Human Error in Medicine*, M. S. Bogner, Ed., pp. 82–98, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1994.
- [6] S. Ternov, "The human side of medical mistakes," in *Error Reduction in Health Care: A Systems Approach to Improving Patient Safety*, P. L. Spath, Ed., pp. 97–138, AHA Press, Chicago, Ill, USA, 2002.
- [7] American Hospital Association, *Hospital Statistics*, American Hospital Association, Chicago, Ill, USA, 1999.
- [8] Centers for Disease Control and Prevention-National Center for Health Statistics, "Births and deaths: preliminary data for 1998," *National Vital Statistics Reports*, vol. 47, no. 25, p. 6, 1999.
- [9] T. S. Lesar, B. M. Lomaestro, and H. Pohl, "Medication-prescribing errors in a teaching hospital: A 9-year experience," *Archives of Internal Medicine*, vol. 157, no. 14, pp. 1569–1576, 1997.
- [10] E. J. Thomas, D. M. Studdert, H. R. Burstin et al., "Incidence and types of adverse events and negligent care in Utah and Colorado," *Medical Care*, vol. 38, no. 3, pp. 261–271, 2000.
- [11] L. T. Kohn, J. M. Corrigan, and M. S. Donaldson, Eds., *To Err Is Human: Building a Safer Health System*, Institute of Medicine, National Academy Press, Washington, DC, USA, 2000.
- [12] BS5760:Part5, *Reliability of Systems, Equipment and Components. Guide to Failure Modes, Effects and Criticality Analysis*, 1991.
- [13] D. Werth, A. Emrich, and A. Chapko, "Prosumerization of mobile service provision: a conceptual approach," *International Journal of Web Portals*, vol. 3, no. 4, pp. 44–55, 2011.
- [14] J. K.-Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.
- [15] A. K. Dey and D. Estrin, "Perspectives on pervasive health from some of the field's leading researchers," *IEEE Pervasive Computing*, vol. 10, no. 2, pp. 4–7, 2011.
- [16] W. Kaiser and M. Sarrafzadeh, "Introduction to special issue on wireless health," *Transactions on Embedded Computing Systems*, vol. 10, no. 1, article 10, 2010.
- [17] S. Deng, C. Youn, Q. Liu, H. Y. Kim, T. Yu, and Y. H. Kim, "Policy adjuster-driven grid workflow management for collaborative heart disease identification system," *Journal of Information Processing Systems*, vol. 4, no. 3, pp. 103–112, 2008.
- [18] M. Lee, J.-W. Lee, K.-A. Kim, and S. S. Park, "Evaluating service description to guarantee quality of U-service ontology," *Journal of Information Processing Systems*, vol. 7, no. 2, pp. 287–298, 2011.
- [19] J. C. Augusto, V. Callaghan, D. Cook, A. Kameas, and I. Satoh, "Intelligent environments: a manifesto," *Human-Centric Computing and Information Sciences*, vol. 3, p. 12, 2013.

- [20] AIAG, A.I.A.G., *Potential Failure Mode and Effects Analysis (FMEA) Reference Manual*, AIAG, Southfield, Mich, USA, 2nd edition, 1995.
- [21] JCAHO, J.C.o.A.o.H.O., Hospital accreditation standards, in Oak Brook Terrace (IL): Joint Commission Resources 2006. pp. 255-256, 261-277.
- [22] P. L. Spath, "Using failure mode and effects analysis to improve patient safety," *AORN Journal*, vol. 78, no. 1, pp. 16-41, 2003.
- [23] M. Ben-Daya and A. Raouf, "A revised failure mode and effects analysis model," *International Journal of Quality & Reliability Management*, vol. 13, no. 1, pp. 43-47, 1996.
- [24] J. B. Bowles, "An assessment of RPN prioritization in a failure modes effects and criticality analysis," *Journal of the IEST*, vol. 47, pp. 51-56, 2004.
- [25] M. Braglia, M. Frosolini, and R. Montanari, "Fuzzy TOPSIS approach for failure mode, effects and criticality analysis," *Quality & Reliability Engineering International*, vol. 19, no. 5, pp. 425-443, 2003.
- [26] C.-L. Chang, P.-H. Liu, and C.-C. Wei, "Failure mode and effects analysis using grey theory," *Integrated Manufacturing Systems*, vol. 12, no. 3, pp. 211-216, 2001.
- [27] W. Gilchrist, "Modelling failure modes and effects analysis," *International Journal of Quality & Reliability Management*, vol. 10, no. 5, pp. 16-23, 1993.
- [28] A. Pillay and J. Wang, "Modified failure mode and effects analysis using approximate reasoning," *Reliability Engineering and System Safety*, vol. 79, no. 1, pp. 69-85, 2003.
- [29] N. R. Sankar and B. S. Prabhu, "Modified approach for prioritization of failures in a system failure mode and effects analysis," *International Journal of Quality & Reliability Management*, vol. 18, no. 3, pp. 324-335, 2001.
- [30] <http://www.palisade.com/>.
- [31] C. Ruggiero, R. Sacile, and M. Giacomini, "Home telecare," *Journal of Telemedicine and Telecare*, vol. 5, no. 1, pp. 11-17, 1999.
- [32] L.-C. Chen, C. W. Chen, Y. C. Weng et al., "An information technology framework for strengthening telehealthcare service delivery," *Telemedicine Journal and e-Health*, vol. 18, no. 8, pp. 596-603, 2012.
- [33] N. Howard and E. Cambria, "Intention awareness: improving upon situation awareness in humancentric environments," *Human-Centric Computing and Information Sciences*, vol. 3, no. 9, 2013.
- [34] M. Brahami, B. Atmani, and N. Matta, "Dynamic knowledge mapping guided by data mining: application on healthcare," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 1-30, 2013.
- [35] L. Prinz, M. Cramer, and A. Englund, "Telehealth: a policy analysis for quality, impact on patient outcomes, and political feasibility," *Nursing Outlook*, vol. 56, no. 4, pp. 152-158, 2008.
- [36] M. S. H. Talpur, "The appliance pervasive of internet of things in healthcare systems," *International Journal of Computer Science Issues*, vol. 10, no. 1, pp. 419-424, 2013.
- [37] J. Basilakis, N. H. Lovell, S. J. Redmond, and B. G. Celler, "Design of a decision-support architecture for management of remotely monitored patients," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 5, pp. 1216-1226, 2010.
- [38] A. Kailas and M. A. Ingram, "Wireless communications technology in telehealth systems," in *Proceedings of the 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology, Wireless (VITAE '09)*, pp. 926-930, May 2009.
- [39] R. D. Berndt, M. C. Takenga, S. Kuehn, P. Preik, G. Sommer, and S. Berndt, "SaaS-platform for mobile health applications," in *Proceedings of the 9th International Multi-Conference on Systems, Signals and Devices (SSD '12)*, pp. 1-4, 2012.
- [40] A. Kuusik, E. Reilent, I. Lõõbas, and M. Parve, "Software architecture for modern telehome care systems," in *Proceedings of the 6th International Conference on Networked Computing (INC '10)*, pp. 326-331, May 2010.

## Research Article

# Energy-Efficient Probabilistic Routing Algorithm for Internet of Things

Sang-Hyun Park,<sup>1</sup> Seungryong Cho,<sup>2</sup> and Jung-Ryun Lee<sup>1</sup>

<sup>1</sup> School of the Electrical Engineering, Chung-Ang University, Seoul 156-756, Republic of Korea

<sup>2</sup> Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Republic of Korea

Correspondence should be addressed to Jung-Ryun Lee; jrlee@cau.ac.kr

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

Academic Editor: Young-Sik Jeong

Copyright © 2014 Sang-Hyun Park 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 the future network with Internet of Things (IoT), each of the things communicates with the others and acquires information by itself. In distributed networks for IoT, the energy efficiency of the nodes is a key factor in the network performance. In this paper, we propose energy-efficient probabilistic routing (EEPR) algorithm, which controls the transmission of the routing request packets stochastically in order to increase the network lifetime and decrease the packet loss under the flooding algorithm. The proposed EEPR algorithm adopts energy-efficient probabilistic control by simultaneously using the residual energy of each node and ETX metric in the context of the typical AODV protocol. In the simulations, we verify that the proposed algorithm has longer network lifetime and consumes the residual energy of each node more evenly when compared with the typical AODV protocol.

## 1. Introduction

Internet of Things (IoT) is a network that enables new forms of communication between people and things and between things themselves. Each of the things or objects in IoT communicates with the others and plays a defined role [1–4]. In the future network with IoT, each node acquires information by itself, and humans only verify the information gathered [5]. IoT can be used in the fields of transportation, healthcare, smart environments, and so forth [1] and key network systems for communicating with things in IoT are radio-frequency identification (RFID) systems, wireless sensor network (WSN), and RFID sensor network (RSN).

In such networks for IoT, nodes are distributed in a certain region for specific purpose and gather the required information, for example, the information about the temperature, motion, and physical changes [6–8]. The nodes forward the gathered information to the intermediate nodes because of the limited transmission range of the node [9, 10]. Therefore, the intermediate nodes use the unintended energy for the packet forwarding of the source node, which induces

high energy consumption of the nodes and thus accelerates *network partitioning*. Therefore, the energy efficiency of the nodes is the key factor that affects the network performance in distributed networks for IoT [11–15].

In addition, relaying information from a source to a destination is one of the most important tasks to be carried out in a large scale and dynamic IoT environment. The typical reactive routing protocols such as ad hoc on-demand distance vector (AODV) and dynamic source routing (DSR) are designed to find just the shortest path [16, 17] without any consideration of the energy consumption of a node. Thus a certain specific node can be selected repeatedly, which may decrease the lifetime of the node and thus cause network partitioning. Also, the reactive routing protocols use the flooding algorithm that forwards route request (RREQ) packets to its all one-hop neighbor nodes to find the routing path. Since excessive RREQ packets lead to mobile node battery run-out [8], it is required to limit the excessive transmission of RREQ packets.

Algorithms to enhance the efficiency of the energy consumption have been widely proposed. In [18], the authors

proposed an algorithm that controls the probability of forwarding RREQ packets according to the residual energy of the node, so the nodes having more residual energy are selected in the routing process. The authors of [19] proposed an energy-efficient routing protocol based on AODV protocol by considering the transmission power and remaining energy capacity of the mobile nodes. However, the above two methods do not consider the link quality of the route, which decreases the network lifetime by wasting the residual energy of the nodes with poor link quality. The authors of [20] proposed probability based improved broadcasting algorithm, which reduces the RREQ messages by using a broadcasting probability together with the consideration of the residual energy of nodes.

On the other hand, most of the current routing protocols use hop count as their route selection metric to find the shortest path between source and destination nodes. However, using only hop count as the routing metric is not appropriate in IoT with dynamic network topology, since it is insensitive to packet loss, data rates, link capacity, link quality, channel diversity, interference, or various other routing requirements. Expected transmission count (ETX) [21] is a metric that aims to provide high throughput, by measuring the packet delivery ratio of the link between neighboring nodes.

In this paper, we propose the energy-efficient probabilistic routing (EEPR) algorithm, which employs both the ETX metric and the residual energy of each node as the routing metrics at the same time. By using the ETX metric, the EEPR algorithm composes the routing path with good link quality. Using the residual energy of each node as a routing metric makes it possible for all the nodes in the network to use their residual energy more evenly. In addition, the EEPR algorithm controls the flooding of RREQ packets in an opportunistic way, so reduces the overhead in the routing process, and finds the energy-efficient routing path more efficiently compared to the typical protocols.

## 2. Proposed Algorithm

The proposed EEPR algorithm controls the request packet forwarding process in order to reduce the packet loss and network congestion in the context of the AODV protocol. A source node that has data packets to transmit forwards the RREQ packets to its one-hop neighbor nodes. In the typical AODV protocol, each node that receives a RREQ packet forwards it to all their one-hop neighbor nodes. On the other hand, a node does not forward the RREQ packet all the time but calculates the forwarding probability via the proposed forwarding probability formula and decides stochastically whether to forward or discard it.

In this paper, we employ two different routing metrics. The first one is the ETX metric which presents the link quality between nodes. In general, probe packets are used to heuristically obtain the ETX value of a link [21]. Each node periodically broadcasts the small-sized probe packets to its one-hop neighbor nodes. The ETX metric is defined as

$$\text{ETX} = \frac{1}{pq}, \quad (1)$$

where  $p$  and  $q$  denote the forward packet delivery ratio and the reverse packet delivery ratio, respectively [21]. Notice that  $p$  and  $q$  are parameters obtained heuristically. Suppose that each node remembers the number of probe packets from the other nodes within  $w$  seconds. When each node periodically broadcasts the probe packets in  $\tau$  cycles, the probe packet delivery ratio of one node at time  $t$  is defined as

$$r(t) = \frac{\text{count}(t-w, t)}{w/\tau}. \quad (2)$$

The denominator of (2) means the number of probe packets that one node has to receive in  $w$  seconds. The numerator of (2) means the number of probe packets that one node receives from  $(t-w)$  seconds to  $t$  seconds. Therefore, from (2), each node can calculate the delivery ratio by counting the number of probe packets. Each node periodically calculates the ETX metric between itself and the neighbor nodes and stores it.

In this paper, we induce the ETX value metric not by using the heuristic method but by using the bit error rate (BER) based on the path-loss model. The received signal strength (RSS), the signal strength that the receiving node senses, is calculated as

$$\text{RSS}_{\text{dB}}(x) = P_{\text{dBm}}^{tx} - P_{\text{dB}}^{\text{loss}}(x), \quad (3)$$

where  $\text{RSS}_{\text{dB}}(x)$ ,  $P_{\text{dBm}}^{tx}$ , and  $P_{\text{dB}}^{\text{loss}}(x)$  are RSS at a node which is away  $x$  km from the source node (dB scale), transmission power of the source node (dBm scale), and path loss at  $x$  km from the source node (dB scale), respectively. Regarding the path loss model, we employ the ITU Ped A channel [22]. Then, signal-to-noise ratio (SNR) is calculated as

$$\text{SNR}(x) = \frac{2 \times \text{RSS}_W(x)}{P_W^{\text{noise}}}, \quad (4)$$

where  $\text{SNR}(x)$ ,  $\text{RSS}_W(x)$ , and  $P_W^{\text{noise}}$  are SNR value at a node which is away  $x$  km from the source node, RSS at a node which is away  $x$  km from the source node (Watt scale), and noise power (Watt scale), respectively. By using the above SNR value, the BER is calculated with the assumption of the ITU Pedestrian A model [22]. Then the desired packet error rate (PER) is obtained as

$$E_{pp} = 1 - (1 - E_b)^{L_{pp}}, \quad (5)$$

where  $E_{pp}$ ,  $E_b$ , and  $L_{pp}$  are PER of a probe packet, BER, and the size of a probe packet, respectively.

We calculate the ETX of each link by counting the number of probe packets that a node receives when the total number of probe packets is 10. The result of the ETX metric via distance is shown in Figure 1.

In this paper, we define  $\text{ETX}_{i-1,i}$  and  $\text{ETX}_{\text{max}}$  as the ETX value between node  $i-1$  and node  $i$  and the maximum ETX value that a link may have, respectively.

The second routing metric to be used in the proposed EEPR algorithm is the residual energy of a node which shows efficiency of the energy consumption in the network. We

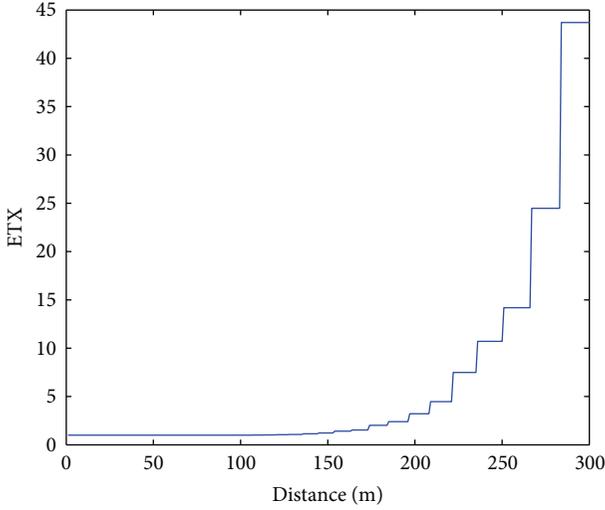


FIGURE 1: ETX metric via distance.

define the residual energy of node  $i$  and maximum residual energy of node  $i$  as  $E_i$ ,  $E_{\max}$ , respectively.

Then, the forwarding probability  $p$  of node  $i$  under the proposed EEPR algorithm is determined by

$$p = \left[ p_{\min} + E_i A \left[ 1 + \frac{(ETX_{i-1,i} - ETX_{\max})}{(1 - ETX_{\max})} \right] \right]^{1/\alpha}, \quad (6)$$

$$A = \frac{1 - p_{\min}}{2 \times E_{\max}},$$

where  $p_{\min}$  and  $\alpha$  are predefined minimum forwarding probability and the weighted factor for variation of the forwarding probability, respectively. From (6), when a node has high residual energy and the link has low ETX value, the forwarding probability is high. Even when a link has far lower ETX value because of good link quality, when the amount of residual energy of a node is small, the forwarding probability is low. Figure 2 shows the forwarding probability as functions of the ETX value and the residual energy when  $ETX_{\max} = 45$ ,  $\alpha = 1$ , and  $p_{\min} = 0.7$ .

When forwarder node  $i$  is set to forward the request packets by using the forwarding probability  $p$ , node  $i$  forwards the request packets to its one-hop neighbor nodes similar to the typical AODV protocol. On the other hand, forwarder node  $i$  is not set to forward the request packet by using the forwarding probability  $p$ , and node  $i$  discards the request packet. An example of this algorithm is shown in Figure 3. When source node  $S$  has data packets to transmit, node  $S$  forwards the RREQ packet to its neighbor nodes 1 and 2. Node 1 has higher residual energy, and the ETX value between node 1 and node  $S$  is good. In this case, node 1 has high forwarding probability. However, node 2 has lower residual energy, and the ETX value between node 2 and node  $S$  is bad. So node 2 has lower forwarding probability.

According to (6), a node with lower residual energy has lower forwarding probability. However, when all nodes in the network have low residual energy, most of forwarder nodes

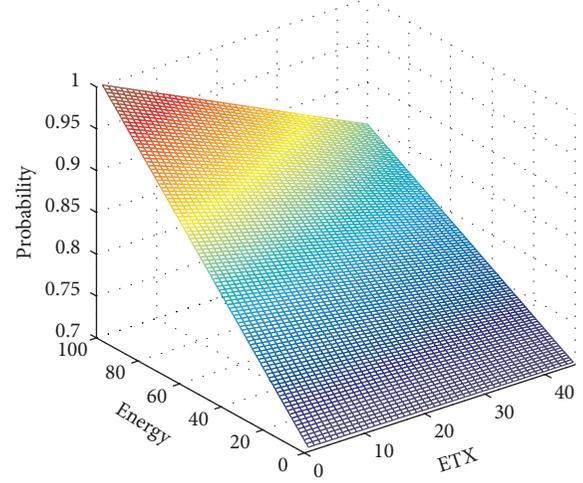


FIGURE 2: Forwarding probability via ETX and residual energy.

discard the RREQ packets because of low forwarding probability. In this case, routing process can be failed continuously.

To solve the above problem, we propose the advanced EEPR algorithm considering both the residual energy of its one-hop neighbor nodes and the average value of residual energy of all nodes in the network. To describe the advanced EEPR algorithm, we should assume two factors. First, it is assumed that each node knows the average value of residual energy of all nodes in the network,  $E_{\text{avg}}$ , which is calculated by the network controller using the periodically received information about the residual energy from each node. Second, each node usually knows the residual energy of its one-hop neighbor nodes from the hello packets which are periodically broadcasted by each node in order to indicate the existence and some information of the node.

The operational procedure of the advanced EEPR algorithm is as follows. When source node needs a routing path, source node broadcasts the RREQ packet to its one-hop neighbor nodes. Then, a forwarder node that receives the RREQ packet calculates forwarding probability  $p$  using its residual energy and ETX value in the EEPR algorithm. However, the node under the advanced EEPR algorithm compares the average value of residual energy of all nodes,  $E_{\text{avg}}$ , with the predefined residual energy threshold,  $E_{\text{th}}$ . If  $E_{\text{avg}}$  is bigger than  $E_{\text{th}}$ , the node regards that the network is in a good energy condition and it is not necessary to make the forwarding probability higher. So, the node calculates the forwarding probability as in (6). If  $E_{\text{avg}}$  is smaller than  $E_{\text{th}}$ , the node thinks that the network is in a low energy condition and tries to make the forwarding probability higher by executing the advanced EEPR algorithm. Each node defines the maximum value of its neighbor node's residual energy as a new  $E_{\max}^{\text{new}}$ , in place of previous  $E_{\max}^{\text{previ}}$ , and calculates the new forwarding probability. So, by using the updated  $E_{\max}^{\text{new}}$  instead of  $E_{\max}^{\text{previ}}$ , we can solve the problem (the forwarding probability is so low that RREQ packets can be

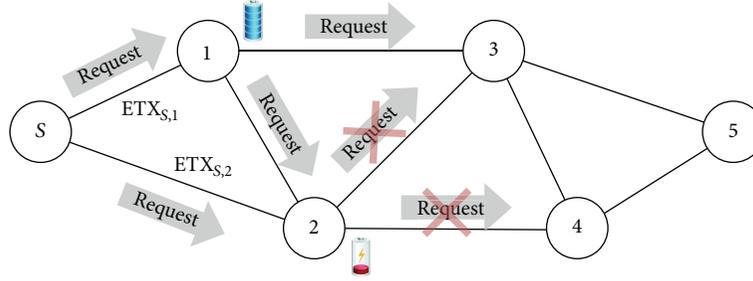


FIGURE 3: Example of the EEPR algorithm.

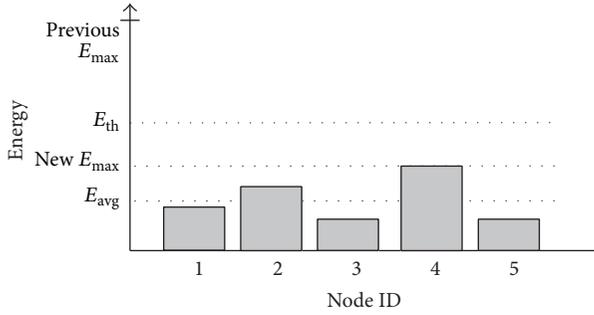


FIGURE 4: Example of the advanced EEPR algorithm.

hardly transmitted to the destination node). This algorithm is described as follows:

$$p = \left[ p_{\min} + E_i A \left[ 1 + \frac{(ETX_{i-1,i} - ETX_{\max})}{(1 - ETX_{\max})} \right] \right]^{1/\alpha},$$

$$A = \begin{cases} \frac{1 - p_{\min}}{2 \times E_{\max}^{\text{previ}}}, & \text{If } E_{\text{avg}} > E_{\text{th}}, \\ \frac{1 - p_{\min}}{2 \times E_{\max}^{\text{new}}}, & \text{If } E_{\text{avg}} \leq E_{\text{th}}. \end{cases} \quad (7)$$

An example of the advanced EEPR algorithm is shown in Figure 4. Node 3 has to calculate the forwarding probability  $p$  and has one-hop neighbor nodes 1, 2, 4, and 5. Before calculating the forwarding probability  $p$ , node 3 has to compare the average value of the residual energy of all the nodes in the network ( $E_{\text{avg}}$ ) with  $E_{\text{th}}$ . In this example,  $E_{\text{avg}}$  is lower than  $E_{\text{th}}$ . Therefore, node 3 has to implement the advanced EEPR algorithm. According to the residual energy of one-hop neighbor nodes, node 4 has the highest residual energy. Therefore, node 3 replaces the previous  $E_{\max}$  value with the residual energy value of node 4. Further, node 3 calculates the forwarding probability  $p$  and determines whether to forward the RREQ packet or not.

In the case of using the advanced EEPR algorithm,  $E_{\text{avg}}$  is the global factor to represent the energy condition of the entire network. In addition, this algorithm considers the residual energy of its one-hop neighbor nodes and updates the value of  $E_{\max}$ . So, the advanced EEPR algorithm considers the energy condition of both local and global networks

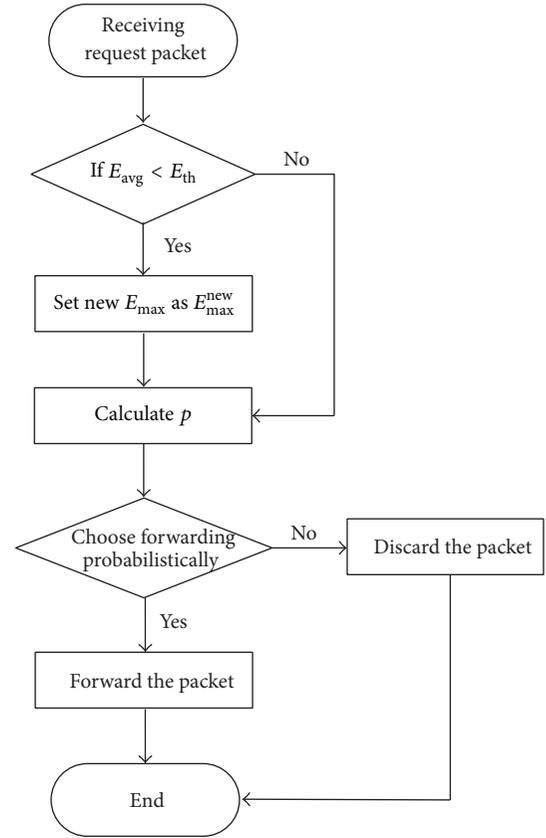


FIGURE 5: Flow chart of the advanced EEPR algorithm.

together. Figure 5 shows the advanced EEPR algorithm in form of a flow chart.

### 3. Simulations and Performance Evaluation

In this paper, we evaluate the performance of the proposed EEPR algorithm and compared it with the typical AODV protocol.

**3.1. Simulation Setup.** Simulations have been performed by the NS-2 simulator version 2.35 on the Linux Fedora 13 [23]. The simulation parameters that are used for the simulation run are listed in Table 1 [24, 25]. In one simulation iteration,

TABLE 1: Factors used in the simulation.

| Simulation factor                  | Value                                           |
|------------------------------------|-------------------------------------------------|
| Topology                           | 1000 m by 1000 m grid random                    |
| Number of nodes                    | 50                                              |
| Path loss model                    | $128.1 + 37.6\log_{10}(\text{dist. (km)})$ (dB) |
| Noise power                        | $10^{-11}$ W                                    |
| Transmission range                 | 300 m                                           |
| Packet size                        | 1,000 bytes                                     |
| Initial node energy                | 10 J~100 J, uniform distribution                |
| Transmission power                 | 0.1 mW                                          |
| Power consumption for transmission | 1.65 W                                          |
| Power consumption for reception    | 1.1 W                                           |
| $E_{\max}$                         | 100                                             |
| $E_{\text{th}}$                    | 40                                              |
| $\text{ETX}_{\max}$                | 45                                              |
| $P_{\min}$                         | 0.7                                             |
| $\alpha$                           | 1                                               |

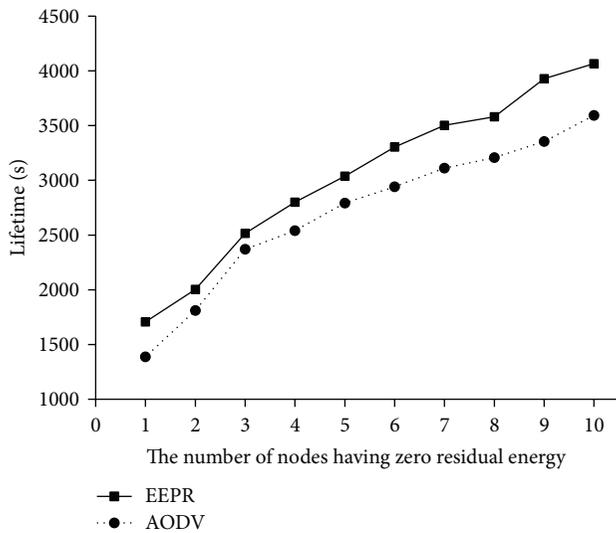


FIGURE 6: Time as a function of the number of nodes having zero residual energy.

the source node requests the routing path and transmits 10 data packets and the size of each packet is 1,000 bytes. This process is iterated 1,500 times and is terminated when all the nodes wear out their residual energy. The initial residual energy of each node is uniformly distributed between 10 J and 100 J.

### 3.2. Performance Evaluation

**3.2.1. Network Lifetime.** Generally the network lifetime is defined as the time difference between the time when the simulation starts and the time when a node having zero residual energy happens. In our work, we extend the concept

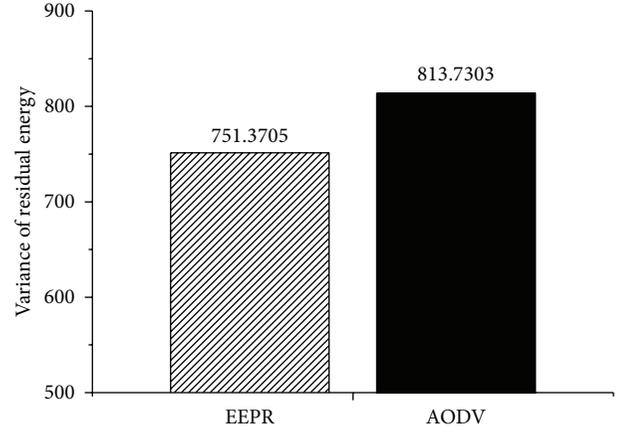


FIGURE 7: Variance of the residual energy.

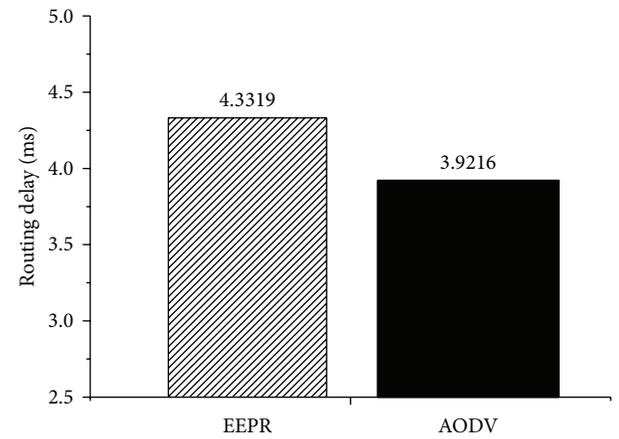


FIGURE 8: Routing setup delay.

of the network lifetime and measure the time between the simulation starting time and the time when  $n$ th node having zero residual energy happens.

Figure 6 shows that the nodes using the EEPR algorithm have approximately 12.57% higher network lifetime when compared with the nodes using the typical AODV protocol. As a result, the EEPR algorithm uses the residual energy of all the nodes in the network more evenly compared with the typical AODV protocol.

**3.2.2. Variance of the Residual Energy.** We measure the residual energy of all the nodes and calculate the variance of the residual energy when the simulation ends. The smaller the variance is, the more evenly the algorithm uses the residual energy of the nodes. For performance comparison, the configuration of residual energy distribution is not changed but fixed regardless of the method used.

The result for the variance of the residual energy of all the nodes in the network is shown in Figure 7. The variance of the residual energy of the nodes under the EEPR algorithm is smaller than that under the typical AODV protocol. The simulation result shows that the nodes under the EEPR

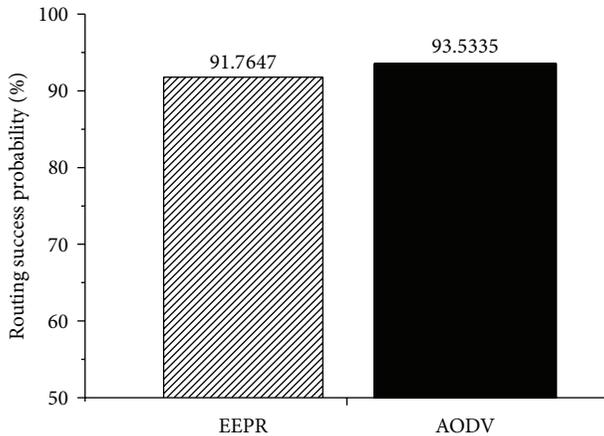


FIGURE 9: Routing success probability.

algorithm spend the residual energy more evenly compared with typical AODV protocol.

**3.2.3. Delay Time in the Routing Process.** Since the EEPR algorithm stochastically controls the number of request packets, the forwarder nodes do not forward the request packets so frequently. This can result in greater routing setup delay compared with the typical AODV protocol. In this paper, we define the routing setup delay as the time difference between the time when a source node forwards the RREQ packets and the time when a destination node receives the first RREQ packet.

The result of the routing setup delay is shown in Figure 8. The routing setup delay under the EEPR algorithm has approximately 0.4 ms higher than that under the typical AODV protocol. It is because the number of forwarded RREQ packets in the network decreases by stochastically controlling the number of the RREQ packets.

**3.2.4. Routing Success Probability.** The EEPR algorithm stochastically controls the number of the RREQ packets. Therefore, as shown in Section 3.2.3, there is a chance that the intermediate nodes on the routing path do not forward the request packets frequently, which may result in the decrease of the routing success probability.

The result for the routing success probability in Figure 9 shows that the routing success probability of the typical AODV protocol is 93.5335%, whereas that of the EEPR algorithm is 91.7647%. It is approximately 1.8% lower than that of the typical AODV protocol, which may be regarded as minor effect.

## 4. Conclusions

In this paper, we proposed EEPR algorithm which employs both the residual energy of a node and the ETX value as the routing metrics, at the same time. The proposed EEPR algorithm stochastically controls the number of the RREQ packets using the residual energy and ETX value of a link on the path and thus facilitates energy-efficient route setup.

Simulation results show that the proposed EEPR algorithm has longer network lifetime and consumes the residual energy of each node more evenly when compared with the typical AODV protocol while the routing setup delay is slightly increased and the routing success probability is slightly decreased.

## 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 Chung-Ang University *Excellent Student Scholarship* in 2012. This research was supported by the Ministry of Knowledge Economy (MKE), Korea, under the Information Technology Research Center (ITRC) support program (NIPA-2012-H0301-12-4004) supervised by the National IT Industry Promotion Agency (NIPA). This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2012RIA2A2A01014170). This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2011-0024132).

## References

- [1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [2] ITU Internet Reports, *The Internet of Things*, 2005.
- [3] O. Vermesan, *Internet of Things Strategic Research Roadmap*, IERC-European, 2011.
- [4] Cluster of European Research Projects on the Internet of Things, *Vision and Challenges for Realising the Internet of Things*, 2010.
- [5] L. Tan and N. Wang, "Future internet: the internet of things," in *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE '10)*, vol. 5, pp. V5376–V5380, chn, August 2010.
- [6] G. Lee, "A cluster-based energy-efficient routing protocol without location information for sensor networks," *International Journal of Information Processing Systems*, vol. 1, no. 1, pp. 49–54, 2005.
- [7] T. Dubey, "Self-localized packet forwarding in wireless sensor networks," *International Journal of Information Processing Systems*, vol. 9, no. 3, 2013.
- [8] A. Sinha, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013.
- [9] X. Li, "Achieving load awareness in position-based wireless ad hoc routing," *KITCS/FTRA Journal of Convergence*, vol. 3, no. 3, 2012.
- [10] C. Huang, "Enhancing network availability by tolerance control in multi-sink wireless sensor networks," *KITCS/FTRA Journal of Convergence*, vol. 1, no. 1, 2010.

- [11] M. Yoon, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *KITCS/FTRA Journal of Convergence*, vol. 4, no. 1, 2013.
- [12] M. M. A. Azim, "MAP: a balanced energy consumption routing protocol for wireless sensor networks," *International Journal of Information Processing Systems*, vol. 6, no. 3, 2010.
- [13] M. S. Obaidat, "DEESR: dynamic energy efficient and secure routing protocol for wireless sensor networks in urban environments," *International Journal of Information Processing Systems*, vol. 6, no. 3, 2010.
- [14] S. Kim, "A hexagon tessellation approach for the transmission energy efficiency in underwater wireless sensor networks," *International Journal of Information Processing Systems*, vol. 6, no. 1, 2010.
- [15] B. Singh, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, no. 13, 2012.
- [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, February 1999.
- [17] B. David Johnson, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996.
- [18] X. Wang, L. Li, and C. Ran, "An energy-aware probability routing in MANETs," in *Proceedings of the IEEE Workshop on IP Operations and Management Proceedings (I POM '04)*, pp. 146–151, October 2004.
- [19] A. P. Patil, "Design of an energy efficient routing protocol for MANETs based on AODV," *International Journal of Computer Science Issues*, vol. 8, no. 4, 2011.
- [20] P. Nand and S. C. Sharma, "Probability based improved broadcasting for AODV routing protocol," in *Proceedings of the International Conference on Computational Intelligence and Communication Systems (CICN '11)*, pp. 621–625, October 2011.
- [21] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.
- [22] ITU-R, *Recommendation M.1225: Guidelines for Evaluation of Radio Transmission Technologies for IMT-2000*, International Telecommunication Union, 1997.
- [23] Information Sciences Institute, "Network Simulator NS2 and Network Animator NAM," <http://www.isi.edu/nsnam>.
- [24] M. Hiyama, "Application of a MANET Testbed for horizontal and vertical scenarios: performance evaluation using delay and jitter metrics," *Human-Centric Computing and Information Sciences*, vol. 1, no. 3, 2011.
- [25] H. H. Choi and J. R. Lee, "A new energy-aware source routing protocol for maximization of network lifetime in MANET," *IEICE Transactions on Information and Systems*, vol. 97, no. 2, pp. 335–339, 2014.

## Research Article

# Usability Analysis of Collision Avoidance System in Vehicle-to-Vehicle Communication Environment

**Hong Cho, Gyoung-Eun Kim, and Byeong-Woo Kim**

*Department of Electrical Engineering, University of Ulsan, No. 93, Daehak-ro, Nam-gu, Ulsan, Republic of Korea*

Correspondence should be addressed to Byeong-Woo Kim; bywokim@ulsan.ac.kr

Received 31 January 2014; Accepted 4 March 2014; Published 10 April 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Hong Cho 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.

Conventional intelligent vehicles have performance limitations owing to the short road and obstacle detection range of the installed sensors. In this study, to overcome this limitation, we tested the usability of a new conceptual autonomous emergency braking (AEB) system that employs vehicle-to-vehicle (V2V) communication technology in the existing AEB system. To this end, a radar sensor and a driving and communication environment constituting the AEB system were simulated; the simulation was then linked by applying vehicle dynamics and control logic. The simulation results show that the collision avoidance relaxation rate of V2V communication-based AEB system was reduced compared with that of existing vehicle-mounted-sensor-based system. Thus, a method that can lower the collision risk of the existing AEB system, which uses only a sensor cluster installed on the vehicle, is realized.

## 1. Introduction

In recent years, with the strict enforcement of safety regulations for vehicles, consumer interest in vehicle safety is growing. Therefore, studies on active safety systems and advanced driver assistance systems (ADASs) have been actively taken up with the aim of ensuring safety, including vehicle control for accident avoidance and mitigation; these features are in contrast with those of conventional passive systems, which ensure safety through simple warnings [1]. One such prominent active system is the autonomous emergency braking (AEB) system. In a recent survey, the European Union (EU) determined that introducing the AEB system could reduce the annual number of deaths and serious injuries in vehicle accidents by more than 8,000 and 20,000, respectively [2]. Generally, an AEB system employs environment-recognition sensors such as radar, lidar, and cameras for detecting risk factors [3, 4]. However, the existing sensor-based systems are able to detect only those vehicles that are within the employed sensors' measurement ranges, and blind spots may occur owing to obstacles. In addition, under bad weather conditions, detection becomes impossible or the detection accuracy drops significantly. For overcoming the limitations

of sensor-based systems, recently, with the advancement of IT technology, cooperative safety system has been introduced. This system is grafted with vehicle safety communication schemes such as vehicle-to-vehicle (V2V) communication and vehicle-to-infrastructure (V2I) communication [5].

Currently, international standards for AEB systems are being formulated worldwide, and various studies on AEB systems are being conducted. The existing studies on AEB systems were conducted based on the performance of sensors employed in vehicles and, therefore, they have limitations concerning detection area [6]. For overcoming these limitations, the current study was conducted based on the cooperative safety system grafted with V2I communication [7]. A limitation of the AEB system, that is, the blind zone occurring at a crossroad, was partially solved through V2I communication by employing radars in traffic lights at crossroads. However, compared with direct V2V communication, V2I communication suffers from real-time limitations and limited detection areas of the sensors installed on the road surface. In the case of V2V communication, studies on the operating environment and evaluations of pure communication technologies were conducted using the existing sensor-based ADAS, and basic studies on cooperative adaptive cruise

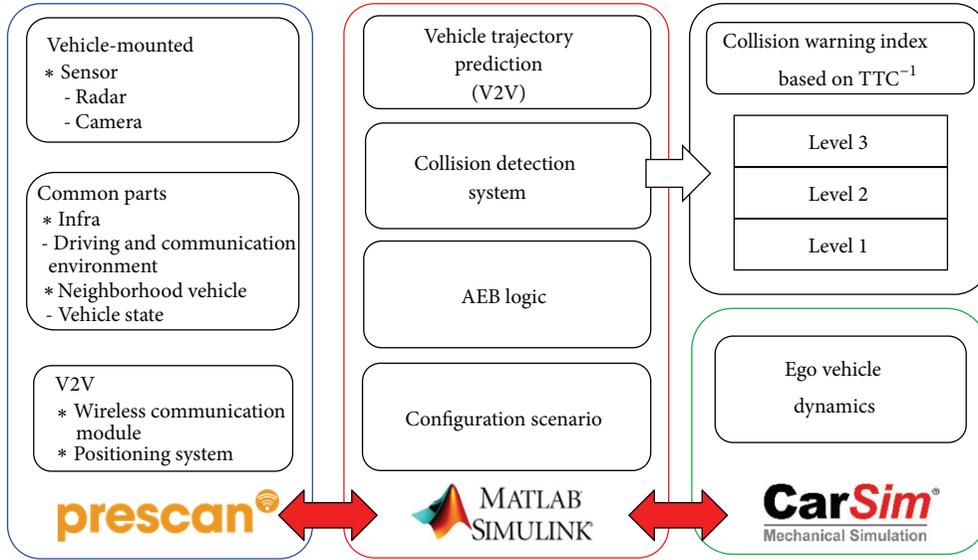


FIGURE 1: Block diagram of analysis model.

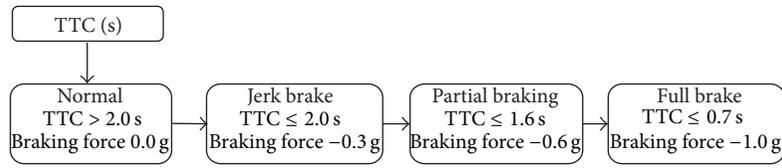


FIGURE 2: AEB logic.

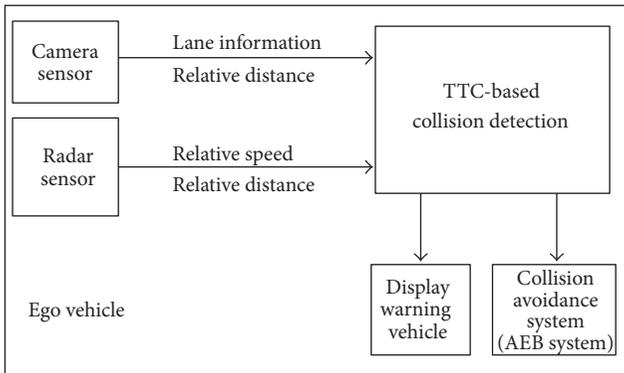


FIGURE 3: Block diagram of vehicle-mounted-sensor-based AEB system.

control were conducted by grafting ADAS with adaptive cruise control and realizing intervehicle communication [8–11]. Consequently, a presentation on the usability of the new conceptual AEB system that employs V2V communication was required.

Therefore, in this study, to overcome the aforementioned limitations of vehicle-mounted-sensor-based systems, we propose a new conceptual AEB system that employs V2V communication along with environment recognition

sensors. In addition, the usability of V2V communication was compared with that of vehicle-mounted sensors by modifying an existing vehicle-mounted-sensor-based AEB system to incorporate V2V communication.

## 2. AEB System Design

**2.1. AEB System Analysis Model.** As shown in Figure 1, for analyzing the usability of V2V communication in comparison with that of vehicle-mounted sensors, a detailed model including various sensors and modules, the driving and communication environment, and vehicle dynamic characteristics was required. Therefore, in this study, PreScan, a commercial simulation code, was used for modeling wireless communication modules, high-precision location measurement systems, the driving and communication environment, and the radar and camera sensors installed in a vehicle. In addition, to determine the dynamic characteristics of a user vehicle equipped with the AEB system, a full car model with multiple degrees of freedom was generated in CarSim, a vehicle dynamic behavior simulation software application, and used. Finally, after interfacing PreScan and CarSim through MATLAB/Simulink, an analysis model was built based on a collision detection system, AEB logic, and the setting of a given scenario.

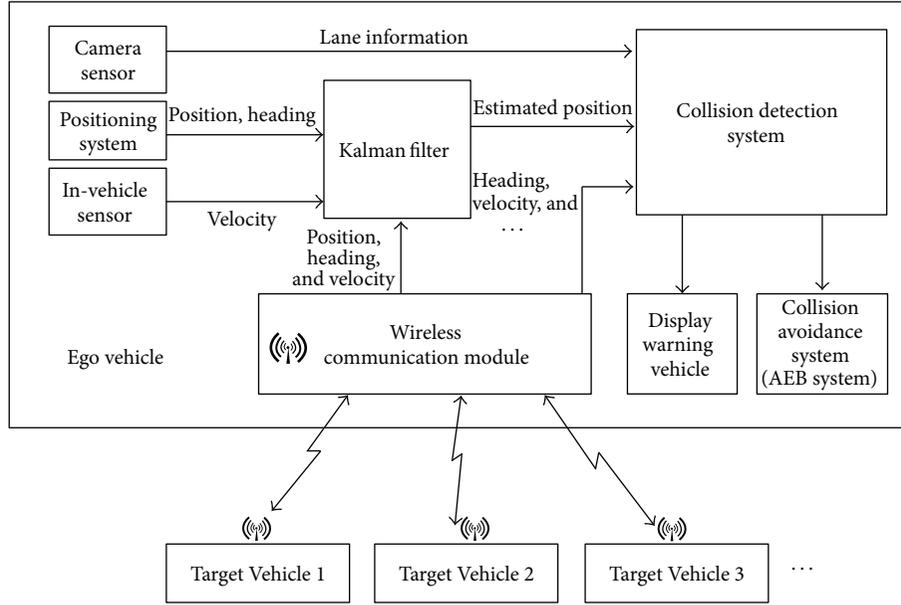


FIGURE 4: Block diagram of V2V communication-based AEB system.

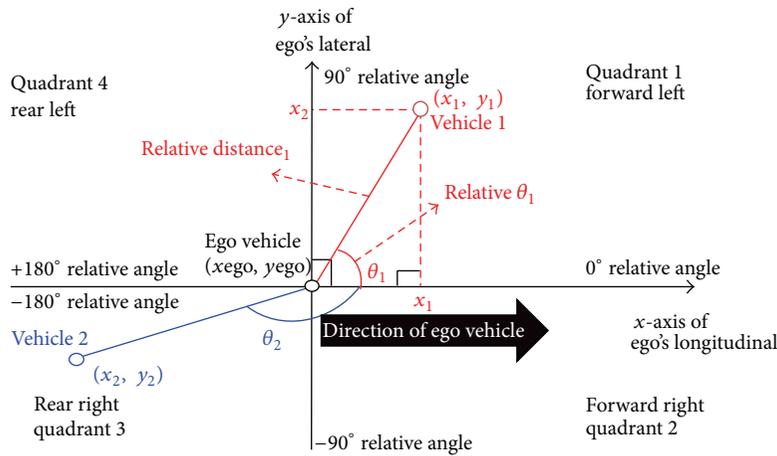


FIGURE 5: Principle of collision detection system.

2.2. AEB System. AEB is an active safety system that measures the degree of risk between a user vehicle and a forward vehicle using vehicle-installed environment recognition sensors such as radars or cameras. It helps in preventing accidents through automatic brake control in risky situations. In Europe, the enforcement of rules concerning AEB systems from 2014 has been initiated. In Europe, the rules concerning AEB systems have come into effect since the beginning of 2014. In 2009, it was proposed that an informal group, called the autonomous emergency braking system (AEBS)/lane departure warning system (LDWS) informal group, will be formed under the Working Party on Brakes and Running Gear (GRRF), a subsidiary body of a World Forum for Harmonization of Vehicle (WP.29), in order to formulate AEBS/LDWS standards. Economic Commission for Europe (ECE) regulations concerning AEBS are being enacted under the United Nations Economic Commission for Europe (UNECE) [12].

TABLE 1: AEB system braking force.

| TTC (s)    | Braking force (g) |
|------------|-------------------|
| $\leq 2.0$ | 0.3               |
| $\leq 1.6$ | 0.6               |
| $\leq 0.7$ | 1.0               |

The AEB system described in this paper meets the required performance specifications defined by the AEB Group. The time of application of automatic braking force by the AEB system was determined based on a collision risk index, that is, time-to-collision (TTC) of the user vehicle and a forward vehicle. TTC can be calculated using (1), which is based on the relative speed and relative distance between the user vehicle and forward vehicle. Table 1 and Figure 2 list and show, respectively, the braking force of the AEB system

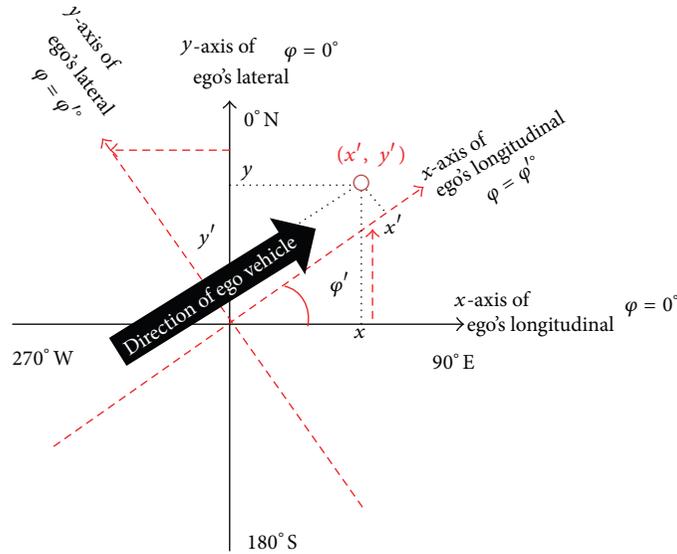


FIGURE 6: CSego transformation with azimuth change.

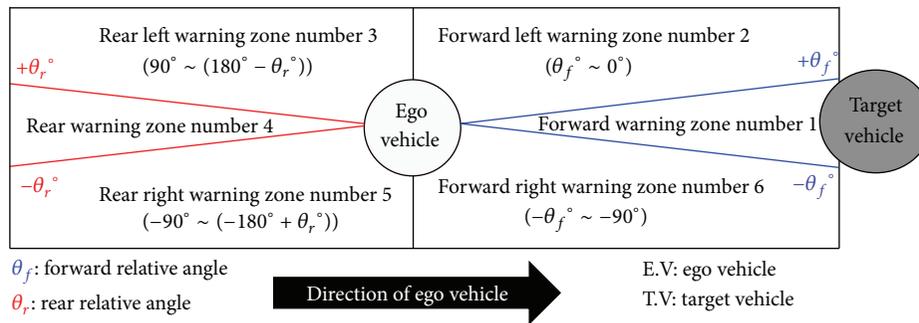


FIGURE 7: Warning zone corresponding to relative angle.

and AEB logic according to changes in TTC; here,  $g$  denotes acceleration due to gravity and it is taken as  $9.8 \text{ m/s}^2$  [7]. Consider

$$\text{TTC (s)} = \frac{\text{Relative distance}}{\text{Relative speed}}. \quad (1)$$

**2.3. Vehicle-Mounted-Sensor-Based AEB System.** Generally, a vehicle-mounted-sensor-based AEB system comprises a camera sensor and long- and short-distance radar sensors. Sensor specifications were determined by referring to the specifications of an actual commercial product. Table 2 lists the specifications of each sensor. The camera sensor mounted in the front provides information about the traffic in a lane and the relative distance to a forward obstacle. Long- and short-distance radar sensors can be used for measuring the distance and speed relative to an obstacle within the forward detection area. In this study, to compensate for the limitations of the camera and radar sensors, the distance and speed relative to a forward obstacle were measured through sensor fusion [13]. The TTC was calculated using the measured information and used as the brake input reference for

the AEB system. Figure 3 shows a block diagram of a vehicle-mounted-sensor-based AEB system.

**2.4. V2V Communication-Based AEB System.** The V2V communication-based AEB system described in this paper was operated based on the collision detection system proposed herein. During operation, the system received information about the nearby vehicles and the user vehicle through V2V communication and employed it for operation. Figure 4 shows the overall system configuration.

First, the location measuring system provides the vehicle location and heading angle information; this system included a noise model corresponding to the tolerance specification (Table 3) of commercial differential global positioning systems (GPS). The received GPS coordinates of the user vehicle and nearby vehicles were expressed in the  $x, y$  coordinate system. The  $x, y$  coordinate system defines a tangent plane in the GPS coordinates of the Infrastystem within 1 km; the  $x$ -axis was defined toward the east and the  $y$ -axis toward the north [14]. Next, various sensors employed in the user vehicle transmit various types of information such as speed, acceleration, and yaw rate through the vehicle's internal

TABLE 2: Vehicle-mounted sensors' specifications.

|                   | Parameter           |                                    |
|-------------------|---------------------|------------------------------------|
| Camera sensor     | Focal length (mm)   | 6                                  |
|                   | FoV (°)             | Azimuth: ±20.5<br>Elevation: ±13.5 |
|                   | Resolution (pixels) | 752 × 480                          |
| Short-range radar | Detection range (m) | 0.2–30                             |
|                   | FoV (°)             | Azimuth: ±40<br>Elevation: ±15     |
| Long-range radar  | Detection range (m) | 2–200                              |
|                   | FoV (°)             | Azimuth: ±10<br>Elevation: ±2.25   |

TABLE 3: Module and sensor specifications of V2V communication-based AEB system.

|                                     | Parameter               |                                    |
|-------------------------------------|-------------------------|------------------------------------|
| Camera sensor                       | Focal length (mm)       | 6                                  |
|                                     | FoV (°)                 | Azimuth: ±20.5<br>Elevation: ±13.5 |
|                                     | Resolution (pixels)     | 752 × 480                          |
| Position system (DGPS)              | Accuracy (cm)           | 50                                 |
| Wireless communication module (V2V) | Communication range (m) | 1,000                              |
|                                     | Message frame           | Basic safety Message (BSM)         |

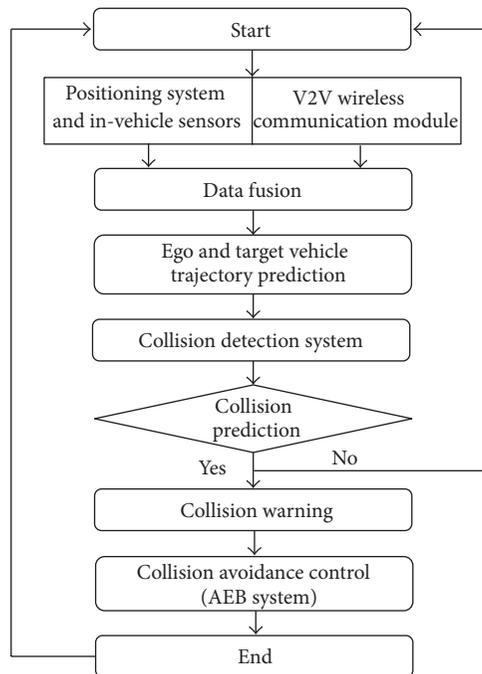


FIGURE 8: Flowchart of V2V communication-based AEB system.

communication system. Finally, the wireless communication module employed for intervehicle communication provides information about the nearby vehicles; a message frame received through the intervehicle communication channel was applied in conjunction with the basic safety message

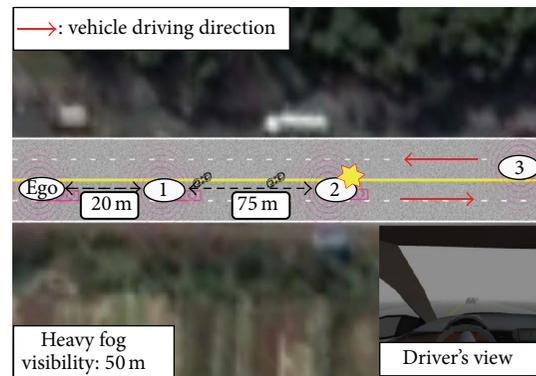
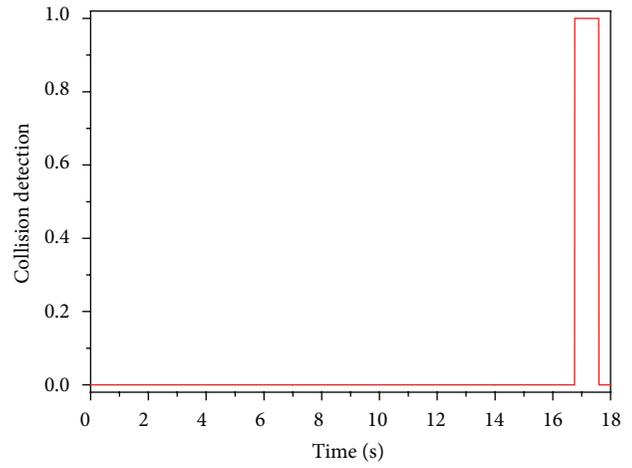


FIGURE 9: Initial road condition.

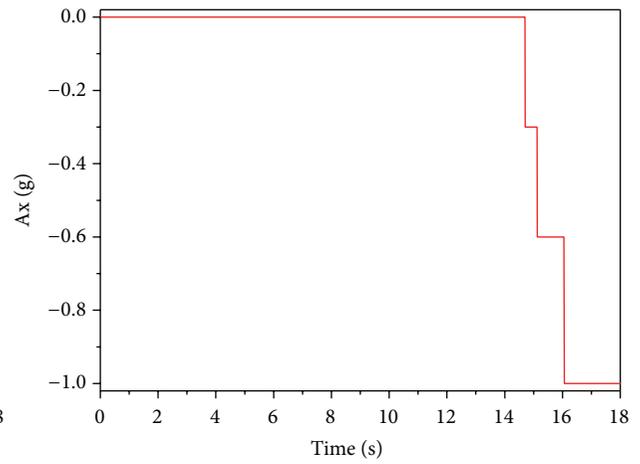
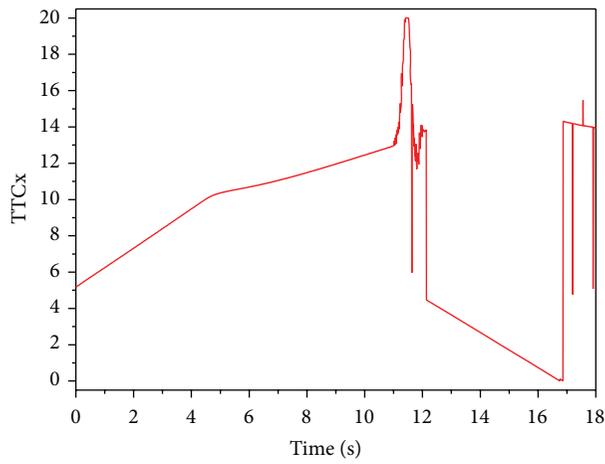
(BSM) standards defined in SAE J2735 [15]. In addition, error of the location measurement systems in the user vehicle and nearby vehicles was calibrated by employing a Kalman Filter; trajectories of the user's and nearby vehicles were measured [16, 17].

As shown in Figure 5, the V2V communication-based collision detection system proposed in this study determines the location of and distance to a nearby vehicle after generating  $C_{Sego}$ , a Cartesian coordinate system, with the current location ( $x_{ego}$ ,  $y_{ego}$ ) of the user vehicle as its origin.  $C_{Sego}$  expresses the longitudinal direction along the  $x$ -axis and the transverse direction along the  $y$ -axis with reference to the user vehicle's driving direction. The locations of nearby vehicles in  $C_{Sego}$  are divided along the quadrants of  $C_{Sego}$  and expressed in relative coordinates ( $x_n$ ,  $y_n$ ) ( $n =$  vehicle id)



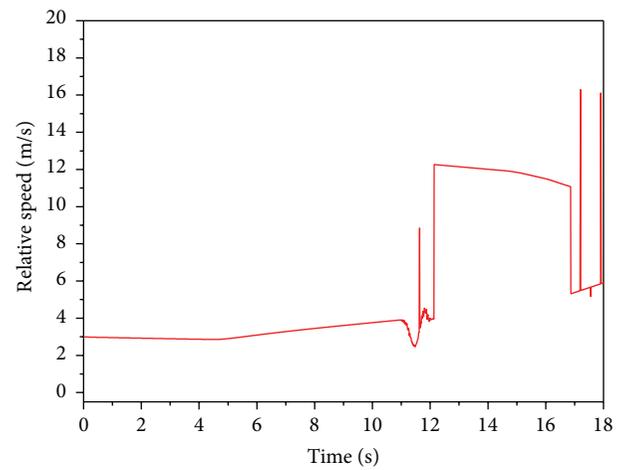
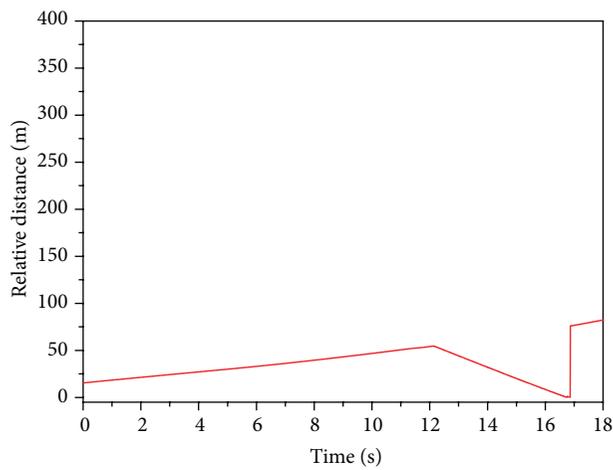
(a) Simulation viewer

(b) Collision detection flag



(c) TTCx

(d) AEB system command



(e) Relative distance

(f) Relative speed

FIGURE 10: Vehicle-mounted-sensor-based AEB system.

after comparing the current location information of the user vehicle and nearby vehicles received through intervehicle communication. The relative angle  $\theta_n$  was calculated by comparison with the azimuth  $\varphi$ , which represents the user vehicle's driving direction. As shown in Figure 5, the nearby vehicle observation system can recognize the locations of nearby vehicles based on the relative angle  $\theta_n$ , which varies along the quadrant. As shown in Figure 6,  $\varphi$  varied with the user vehicle's heading angle and was set to be  $0^\circ$  with respect to the east direction.

As shown in Figure 6, the coordinate axis of CSego rotated according to changes in  $\varphi$ . The longitudinal driving direction of the user vehicle was always matched with the  $x$ -axis using the rotational transformation matrix equation (2), which considers the coordinate axis' rotation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \quad (2)$$

As shown in Figure 7, the warning zone was determined based on the relative angle obtained following the above process.  $\theta_f$  and  $\theta_r$  are the error ranges of the relative angle in the case that the user vehicle and all vehicles in the same traffic lane move in the middle of the traffic lane.

The relative angle, relative distance, and heading angle of the nearby vehicle are the parameters that determine the warning zone. The heading angle is an important parameter for determining the nearby vehicle's driving direction.

The relative heading angle between the user vehicle and the nearby vehicle was obtained for determining whether the nearby vehicle is driving in the same direction as the user vehicle, entering a crossroad, or driving in the opposite traffic lane. In addition, the relative distance can be obtained using the user vehicle's barycentric coordinate system CSego. However, the relative distance obtained thus does not account for the size of the vehicle; therefore, the relative distance was calibrated assuming a circular vehicle [18]. The TTC was calculated based on the obtained relative distance and relative speed of the user vehicle and the nearby vehicle, and the calculated TTC was used as the collision risk index for the AEB system. Figure 8 shows a flowchart of the V2V communication-based AEB system.

### 3. Simulation and Results

**3.1. Simulation Scenario.** As shown in Figure 9, the driving direction and scenarios were defined for a comparative analysis of the vehicle-mounted-sensor-based and V2V communication-based AEB systems. The driving direction was determined based on the conditions under which accidents generally occur. This includes the condition of heavy fog, under which visibility is less than 50 m, which makes it difficult for a driver to recognize forward risk situations.

An ego vehicle mounted with an AEB system can avoid and mitigate the effects of collisions in the longitudinal direction. Consider the following scenario. The driver of Vehicle 1 changes lanes to avoid collision after finding that Vehicle 2 in front is stationary. Vehicle 3 is driving in the opposite traffic lane. The simulation scenario is summarized in Table 4.

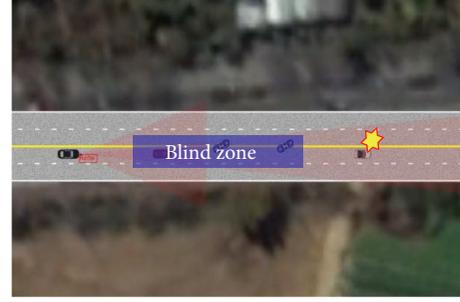


FIGURE 11: Limitations of vehicle-mounted sensor.

TABLE 4: Simulation scenario.

| Vehicle   | Initial speed | End speed | Note          |
|-----------|---------------|-----------|---------------|
| Ego       | 50 km/h       | 60 km/h   | AEB system    |
| Vehicle 1 | 60 km/h       | 60 km/h   | Lane change   |
| Vehicle 2 | 55 km/h       | 0 km/h    | Vehicle fault |
| Vehicle 3 | 70 km/h       | 70 km/h   | Opposite lane |

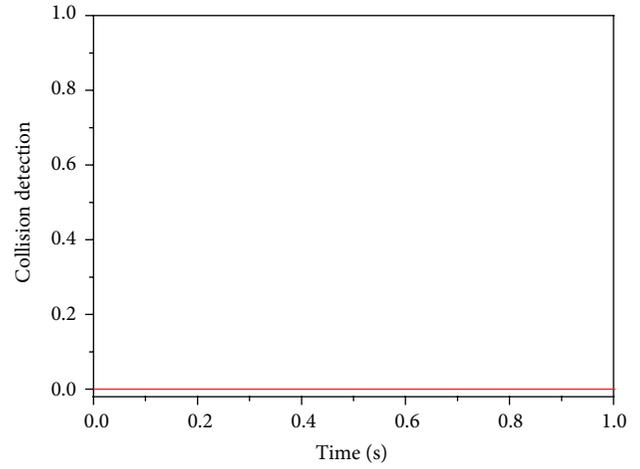
**3.2. Simulation Results.** Simulation was performed by employing the vehicle-mounted-sensor-based AEB system and the V2V communication-based AEB system in the scenarios defined in Table 4.

Figure 10 shows the simulation results of the vehicle-mounted-sensor-based AEB system. The system was capable of detecting forward vehicles only, such as the ego in this case, located in the traffic lane. Therefore, the longitudinal collision risk index, TTCx, shown in Figure 10(c), increased gradually from 0 s to about 11 s and then decreased rapidly. This is because Vehicle 1, which was running initially on the same traffic lane, changed lanes owing to the detection of a stationary vehicle; the sensor in the Ego vehicle detected a stationary vehicle (Vehicle 2) on the same traffic lane. As shown in Figure 10(d), it can be seen that the AEB system was applied normally with braking force as TTCx varied. However, it can be confirmed from the relative distance graph in Figure 10(e) that collision was predicted when the relative distance changed to 0 m. In fact, even in the simulation environment, the occurrence of a collision can be confirmed based on the vehicle driving state shown in Figure 10(a) and the collision detection flag shown in Figure 10(b). In addition, a comparison of the relative speed before the time (about 14.8 s) of braking force application by the AEB system, and the relative speed at the time of collision, shown in the relative speed graph of Figure 10(f), indicates that the speed was reduced by about 1.8 km/h (0.5 m/s). Therefore, it can be said that the collision avoidance relaxation rate achieved with the vehicle-mounted-sensor-based AEB system in relevant scenarios was not more than 3%. This can be ascribed to the vehicle-mounted sensors' inability to detect the stationary vehicle (Vehicle 2) on the road ahead owing to the presence of a blind zone due to a front vehicle, as shown in Figure 11.

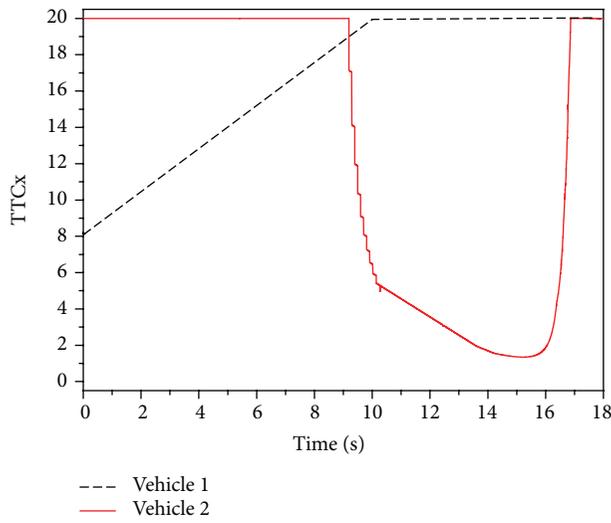
In contrast, in the case of the V2V communication-based AEB system shown in Figures 12(a) and 12(b), there was no collision between the Ego vehicle and the stationary vehicle (Vehicle 2). The TTCx results shown in Figure 12(c) indicate



(a) Simulation viewer

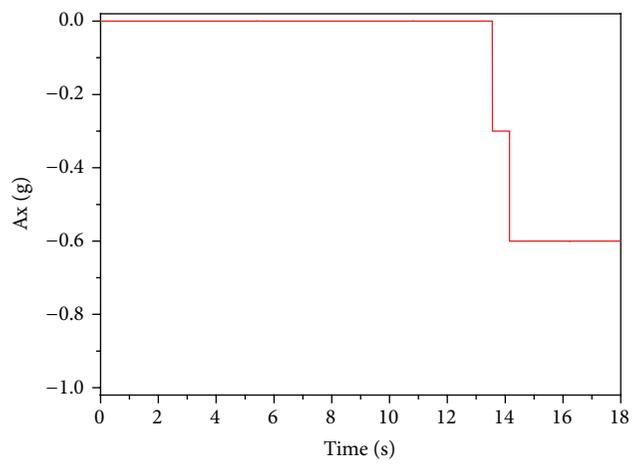


(b) Collision detection flag

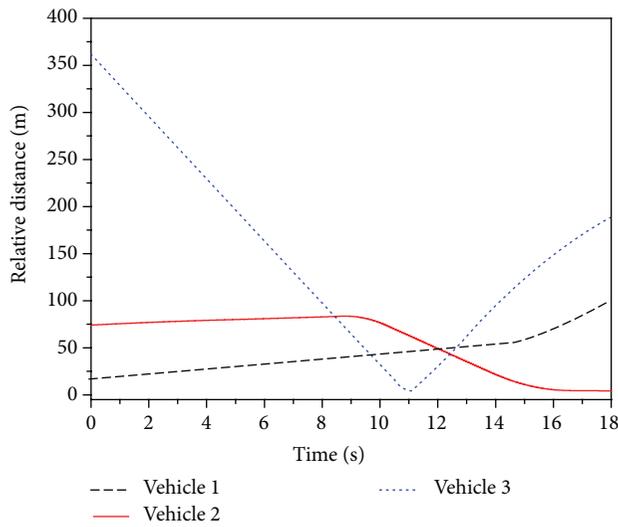


-- Vehicle 1  
— Vehicle 2

(c) TTCx

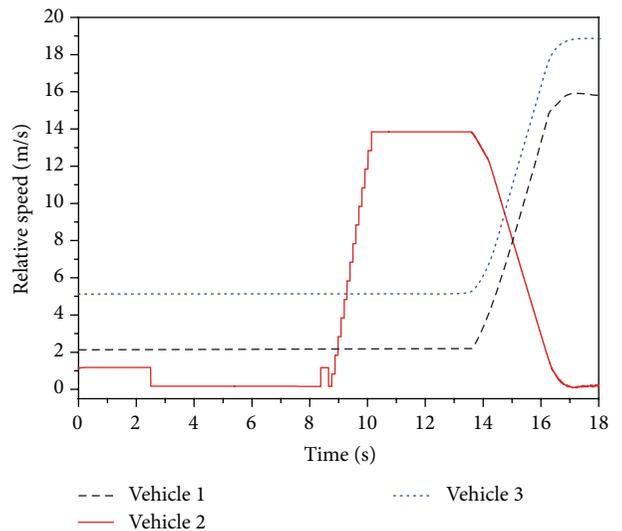


(d) AEB system command



-- Vehicle 1                      ····· Vehicle 3  
— Vehicle 2

(e) Relative distance



-- Vehicle 1                      ····· Vehicle 3  
— Vehicle 2

(f) Relative speed

FIGURE 12: V2V communication-based AEB system.

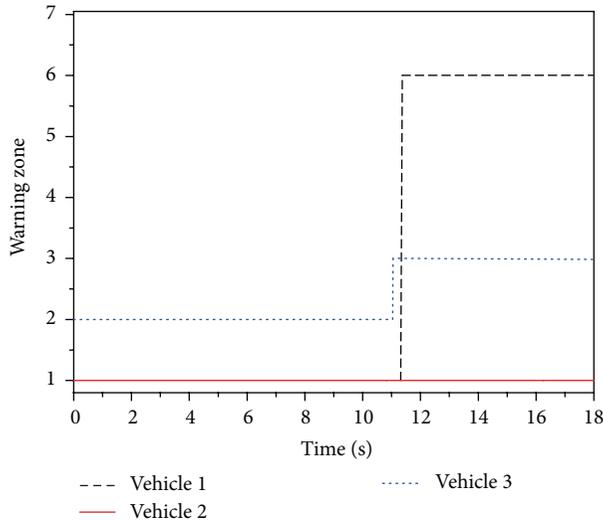


FIGURE 13: Warning zone change during simulation.

that there was a collision risk in the longitudinal direction with Vehicles 1 and 2, which were located in front of the Ego vehicle. However, Vehicle 3 was not represented in the TTCx graph. Vehicle 3 was determined to be a vehicle in the opposite traffic lane based on heading angle information received through V2V communication and was excluded by the collision detection system. It can be inferred from Figure 12(d) that braking force can be applied stably based on changes in the TTCx calculated by the collision detection system after predetecting the forward stationary vehicle (Vehicle 2) through intervehicle communication. In addition, the plots of relative distance (Figure 12(e)) and relative speed (Figure 12(f)) indicate that the collision avoidance relaxation rate reached 100% with the avoidance of collision because the relative speed decreased to 0 m/s before the relative distance to Vehicle 2 decreased to 0 m. In addition, it can be seen from Figure 12(f) that the relative speed with respect to Vehicle 2 increased slowly after about 9 s, indicating that the vehicle became stationary at about 9 s.

Figure 13 shows the warning zone according to simulation time. It can be seen that Vehicle 1, which was running in front of the Ego vehicle, changed its traffic lane after detecting a stationary vehicle. Thus, the warning zone of Vehicle 1 was changed from 1 to 6. Vehicle 2 was the stationary vehicle and there was no change in its traffic lane; thus, there was no change in its warning zone. Finally, Vehicle 3 was driving on the opposite traffic lane, and its warning zone changed from 2 to 3 at about 11.5 s because Vehicle 3 overtook the user vehicle. This can also be seen in the relative distance graph (Figure 12(e)), which shows that the relative distance with respect to Vehicle 3 was closer to 0 at about 11.5 s and started increasing thereafter.

#### 4. Conclusions

In this study, the usability of the proposed V2V communication-based AEB system was compared with that of the existing vehicle-mounted-sensor-based system.

An analysis model was built for determining the usability of the V2V communication-based AEB system. The analysis model considered the vehicle-mounted sensor and V2V communication environments. Furthermore, the existing vehicle-mounted-sensor-based AEB system was realized using this model. In addition, a new conceptual AEB system was proposed and developed by combining V2V communication technology with environment-recognition sensors. Then, a comparative analysis simulation of the V2V communication-based AEB system versus the vehicle-mounted-sensor-based system was conducted in the same scenario. The simulation results show that in the case of the existing vehicle-mounted-sensor-based AEB system, braking application time lengthened and a collision occurred owing to the system's detection area limitation. However, in the case of the V2V communication-based AEB system, collision was avoided regardless of driving conditions and obstacles through collision risk detection within the range of intervehicle communication. In addition, in the case of the existing vehicle-mounted-sensor-based AEB system, the collision avoidance relaxation rate was no more than 3%. In contrast, in the case of the V2V communication-based AEB system, the collision avoidance relaxation rate reached 100%.

Therefore, the usability of the V2V communication technology was demonstrated through the aforementioned comparative analysis. Future studies will be aimed at testing the proposed system in the V2V communication environment with an actual vehicle used in practice and analyzing the proposed system in various scenarios and driving environments.

#### 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 Ministry of Science, ICT, and Future Planning (MSIP), Korea, under the Convergence Information Technology Research Center (C-ITRC) support program (NIPA-2013-H0401-13-1008) supervised by the National IT Industry Promotion Agency (NIPA). And this paper is an extended and improved version of the paper accepted for the KCIC-2013/FCC-2014 Conferences.

#### References

- [1] M. Lu, K. Wevers, R. Van Der Heijden, and T. Heijer, "ADAS applications for improving traffic safety," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '04)*, vol. 4, pp. 3995–4002, October 2004.
- [2] Economic Commission for Europe, "An introduction to the new vehicle safety regulation," Tech. Rep. WP.92-145-08, Informal Document, Geneva, Switzerland, 2008.
- [3] K. Goswami, G. S. Hong, and B. G. Kim, "A novel mesh-based moving object detection technique in video sequence," *Journal of Convergence*, vol. 4, no. 3, pp. 20–24, 2013.

- [4] Verma, V. Jain, and R. Gumber, "Simple fuzzy rule-based edge detection," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 575–591, 2013.
- [5] D. Caveney and W. B. Dunbar, "Cooperative driving: beyond V2V as an ADAS sensor," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV '12)*, pp. 529–534, June 2012.
- [6] N. Kaempchen, B. Schiele, and K. Dietmayer, "Situation assessment of an autonomous emergency brake for arbitrary vehicle-to-vehicle collision scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 4, pp. 678–687, 2009.
- [7] J. Lee, S. Jo, J. Kwon, T. Hong, and K. Park, "Development of V2I-based Intersection Collision Avoidance System," in *Proceedings of the Conference for Korea Institute of ITS*, pp. 90–96, May 2013.
- [8] G. Peng, K. Zeng, and X. Yang, "A hybrid computational intelligence approach for the VRP problem," *Journal of Convergence*, vol. 4, no. 2, pp. 1–4, 2013.
- [9] S. Wang, A. Huang, and T. Zhang, "Performance evaluation of IEEE 802.15.4 for V2V communication in VANET," in *Proceedings of the International Conference on Computational and Information Sciences*, pp. 1603–1606, June 2013.
- [10] Z. Taysi and A. Yavuz, "ETSI compliant GeoNetworking protocol layer implementation for IVC simulations," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, p. 4, 2013.
- [11] C. Desjardins and B. Chaib-Draa, "Cooperative adaptive cruise control: a reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1248–1260, 2011.
- [12] Economic Commission for Europe, *Automatic Emergency Braking and Lane Departure Warning Systems*, UN/ECE/TRANS/WP29/GRRF-65-Inf19, Informal Group on Automatic Emergency Braking and Lane Departure Warning System, Geneva, Switzerland, 2009.
- [13] J. S. Sangorrin, J. Sparbert, U. Ahlrichs, W. Branz, and O. Schwindt, "Sensor data fusion for active Safety Systems," *SAE International Journal of Passenger Cars—Electronic and Electrical Systems*, vol. 3, no. 2, pp. 154–161, 2010.
- [14] "PreScan R6. 6. 0 Help Manual," May 2013.
- [15] C. Hedges and E. Perry, "Overview and use of SAE J2735 message sets for commercial vehicles," SAE Technical Paper 2008-01-2650, October 2008.
- [16] J. Huang and H.-S. Tan, "A low-order DGPS-based vehicle positioning system under urban environment," *IEEE/ASME Transactions on Mechatronics*, vol. 11, no. 5, pp. 567–575, 2006.
- [17] S. Ammoun, F. Nashashibi, and C. Laurgeau, "Real-time crash avoidance system on crossroads based on 802.11 devices and GPS receivers," in *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC '06)*, pp. 1023–1028, September 2006.
- [18] Y. Wang, "Vehicle collision warning system and collision detection algorithm based on vehicle infrastructure integration," in *Proceedings of the 7th Advanced Forum on Transportation of China (AFTC '11)*, pp. 216–220, October 2011.

## Research Article

# An Efficient and Secure *m*-IPS Scheme of Mobile Devices for Human-Centric Computing

Young-Sik Jeong,<sup>1</sup> Jae Dong Lee,<sup>2</sup> Jeong-Bae Lee,<sup>3</sup> Jai-Jin Jung,<sup>4</sup> and Jong Hyuk Park<sup>2</sup>

<sup>1</sup> Department of Multimedia Engineering, Dongguk University, Seoul 100-715, Republic of Korea

<sup>2</sup> Department of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul 139-743, Republic of Korea

<sup>3</sup> Department of Computer Engineering, Sun Moon University, Asan 330-150, Republic of Korea

<sup>4</sup> Department of Multimedia Engineering, Dankook University, Cheonan 330-714, Republic of Korea

Correspondence should be addressed to Jong Hyuk Park; parkjonghyuk1@hotmail.com

Received 14 October 2013; Accepted 4 January 2014; Published 23 March 2014

Academic Editor: Jianhua Ma

Copyright © 2014 Young-Sik Jeong 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.

Recent rapid developments in wireless and mobile IT technologies have led to their application in many real-life areas, such as disasters, home networks, mobile social networks, medical services, industry, schools, and the military. Business/work environments have become wire/wireless, integrated with wireless networks. Although the increase in the use of mobile devices that can use wireless networks increases work efficiency and provides greater convenience, wireless access to networks represents a security threat. Currently, wireless intrusion prevention systems (IPSs) are used to prevent wireless security threats. However, these are not an ideal security measure for businesses that utilize mobile devices because they do not take account of temporal-spatial and role information factors. Therefore, in this paper, an efficient and secure mobile-IPS (*m*-IPS) is proposed for businesses utilizing mobile devices in mobile environments for human-centric computing. The *m*-IPS system incorporates temporal-spatial awareness in human-centric computing with various mobile devices and checks users' temporal spatial information, profiles, and role information to provide precise access control. And it also can extend application of *m*-IPS to the Internet of things (IoT), which is one of the important advanced technologies for supporting human-centric computing environment completely, for real ubiquitous field with mobile devices.

## 1. Introduction

Rapid developments in wireless and mobile IT technologies have led to their application in many real-life areas, such as disasters, home networks, mobile social networks, medical services, industry, schools, and the military. In today's contemporary information-oriented society, mobile devices are increasingly being utilized in diverse business and social areas of life. Business/work environments have become wire/wireless, integrated with wireless networks. Although the increase in the use of mobile devices that can use wireless networks increases work efficiency and provides greater convenience, wireless access to networks represents a security threat.

In wired networks, solutions such as intrusion prevention systems (IPSs), intrusion detection systems (IDSs), and firewalls are used to prevent illegal external access. In wireless network environments, wireless IPSs are used to reinforce security against accessing mobile devices. Wireless IPSs have also been developed to prevent security threats that may occur in wireless environments. However, they are vulnerable to attack because they use pattern-based engines to detect potential attacks. Furthermore, business/work environments are too diversified to enable accurate detection and prevention of attacks. In essence, the same rules cannot be applied to all mobile business devices because of considerations of time, space, and individual roles [1, 2].

TABLE 1: Characteristics of MAC, DAC, and RBAC.

|                    | MAC                                                                           | DAC                                       | RBAC                        |
|--------------------|-------------------------------------------------------------------------------|-------------------------------------------|-----------------------------|
| Access authority   | System                                                                        | Owner                                     | Central authority           |
| Criteria of access | Security level                                                                | Identity                                  | Role                        |
| Strategy           | Stiff                                                                         | Flexible                                  | Flexible                    |
| Merits             | Secure                                                                        | Easy implementation,<br>flexible response | Easy management             |
| Demerits           | Difficult implementation and<br>management, high cost, and low<br>performance | Possible illegal behavior                 | Existence of conflict roles |

In this wireless IPS study, trends and related security threats and requirements in mobile business work environments are discussed. An efficient and secure mobile-IPS (*m*-IPS) is proposed for businesses utilizing mobile devices in mobile environments for human-centric computing, which should provide the ease of utility of mobile devices for protecting from threats. The *m*-IPS system incorporates temporal-spatial awareness in human-centric computing with various mobile devices and checks users' temporal-spatial information, profiles, and role information to provide precise access control. And it also can extend application of *m*-IPS to the Internet of things (IoT), which is one of the important advanced technologies for supporting human-centric computing environment completely, for real ubiquitous field with mobile devices.

This paper is organized as follows. In Section 2, we discuss related works: research trends of access control. We discuss wireless security threats and requirements in Section 3. We explain the detailed proposed scheme: *m*-IPS scheme including main concept, system architecture, service scenario, and evaluation in Section 4. Finally, the conclusion should be provided in Section 5.

## 2. Related Work

Access control techniques are traditionally subdivided into mandatory access control (MAC), discretionary access control (DAC), and role-based access control (RBAC). In MAC, only the administrator has access and directly controls network access by other users after checking access classes. This technique has disadvantages such as the following: control is difficult if the numbers of users increase or if there are diverse access classes. Therefore, it is not well suited to commercial applications. DAC regulates access to objects based on the identity of the subjects or the organizations to which they belong. With DAC, users can illegally pass on access permissions to other individuals or groups. RBAC allows users to access information related to the specific roles that they have been assigned. In real-life settings, RBAC may not be appropriate because of conflicts that may occur between roles [2–5]. The characteristics, merits, and demerits of access control techniques are set forth in Table 1.

Wireless IDSs monitor the radio spectrum for the existence of illegal access points (APs) and malicious devices

and the presence of wireless attack tools. In general, wireless IPSs refer to systems that implement not only detection but also prevention based on the level of risks after automatic classification. These systems aim at preventing unauthorized access to local area networks of wireless devices and other information assets. A wireless IPS is composed of a server, a database, sensors, and a console. The server collects raw data from multiple sensors and analyzes the collected data. The database is used to store information obtained from the sensors and servers. The sensors monitor wireless signals and the information obtained from the server. The console provides an interface for the administrator and users who need information from the server or sensors [6, 7].

Chen et al. proposed a wireless IPS framework using signature detection rules based on specific device information that can reduce false-positive rates and intelligent prior attack recognition engines that can predict and prevent attackers [1]. Silas et al. described wireless security threats and a method to respond to these threats through wireless IPSs [7]. Nyanchama and Osborn proposed a common framework for wireless IPSs and described core technologies used in the framework [6]. Kirkpatrick et al. proposed a wireless IDS using short message service technology, which proactively detects common wireless attacks, such as WEP cracking, MAC address spoofing, and war driving [8]. Timofte proposed a wireless transport layer-based IPS model that can detect and block user traffic through a logical single path between all wireless devices and the destination [9]. Zhang et al. described four major blocking techniques in wireless networks and assessed their blocking performances by a device manufacturer based on test beds for these methods and discussed the implications of their experimental results for wireless IPS designs [10]. Hsieh et al. proposed a model for wireless attack detection and prevention using honey pot-based intelligent prior attack recognition engines and tried to minimize false-positive rates using this model [11]. Tahir [12] provides our understanding of domain and introduced spatial domain roles. It is emphasized that purpose should be attached to spatial roles that should be represented within organizational domains that may have multilevel and multidomain relationships. And it is also shown how our extended RBAC model can make use of the notion of spatial domain to allow administrator to flexibly partition the objects according to geographical boundaries.

### 3. Wireless Security Threats and Requirements

In this chapter, wireless security threats and requirements are discussed in detail prior to constructing a temporal-spatial awareness-based efficient  $m$ -IPS scheme to enable more secure use of mobile devices in business and social life.

**3.1. Threats in Wireless Security.** General wireless security threats that may occur in business and social life using mobile devices include rogue evil twin APs, ad hoc networks, RF jamming, deauthentication, MAC spoofing, WEP key cracking, and sniffing [1, 2]. Table 2 classifies wireless security threats with respect to confidentiality, integrity, and availability (three elements of security) that may occur in business and social life using mobile devices.

The following are examples of security threats that may additionally occur in business and social life using mobile devices if temporal-spatial and role elements are not considered in wireless IPSs.

*Case 1.* When temporal-spatial elements are not considered: in an office environment using mobile devices that provide service to users, logs may increase rapidly due to floating populations. This will cause system overloads and adversely affect the ability of the wireless IPS to detect illegal devices and judge the level of threat. If mobile device security is necessary in nonpermanent spaces, such as meeting rooms at particular times, wireless device detection and blocking based on uniform rules will be difficult.

*Case 2.* When roles are not considered: existing wireless IPSs use access control lists (ACLs) of user names and groups to provide mobile device security. However, ACLs cannot detect malicious acts that are carried out by devices registered: on the so-called white list of privileged users. For instance, an attacker could acquire the device of a finance department staff member with diverse access rights and then bypass the firewall and wirelessly access the server of this department to revise, copy, or delete files. Access cannot be prevented because the request will have come from a device viewed as secure.

**3.2. Requirements for Wireless Security.** To prevent wireless security threats in business and social life using mobile devices, three elements of security, confidentiality, integrity, and availability, plus access control based on temporal-spatial and role elements, are required. With regard to confidentiality, wireless signals can be propagated to many unspecified users in mobile offices, and sensitive data, such as personal information and financial details, stored in wireless terminals are quite likely to be leaked. Wireless terminals are more vulnerable than wired terminals to security attacks from wireless sniffing and evil twin APs. All businesses have to take steps to help prevent such attacks [1, 6, 7, 10, 11]. Regarding integrity, the so-called Man in the Middle attacks may cause system failure and work confusion. These involve illegal changes and deletions in data and forged data insertion during wireless communication transmission between mobile

TABLE 2: Classification of wireless security threats during business work using mobile devices.

| Threat classification | Confidentiality | Integrity | Availability |
|-----------------------|-----------------|-----------|--------------|
| Rogue AP              | o               | o         | —            |
| Ad hoc network        | o               | —         | —            |
| Evil twin/honeypot AP | o               | o         | —            |
| RF jamming            | —               | —         | o            |
| Deauthentication      | —               | —         | o            |
| MAC spoofing          | o               | o         | —            |
| WEP key cracking      | o               | o         | —            |
| Sniffing              | o               | —         | —            |

o: effect, —: no effect.

devices. Security measures that can guarantee integrity during data transmission in wireless spaces are necessary [1, 6, 7, 10, 11]. With respect to availability, Denial of Service attacks damage system availability and productivity, thereby reducing system resources and accessibility to information. Therefore, in business and social environments that depend on wireless communication using mobile devices, measures are required to prevent RF jamming on layer 1 and layer 2 of Open System Interconnection (7 layers) systems and to prevent attacks, such as DoS using deauthentication packets [1, 13–15].

To reduce false-positive rates and system loads in business and social settings where diverse mobile devices, roles, and environments exist, the mobile-intrusion prevention system ( $m$ -IPS) is needed to ensure better access control based on temporal, spatial, and contextual roles for efficiency and security. The access control should be able to respond to diverse exceptions that may occur in offices [2, 5, 6].

### 4. $m$ -IPS Scheme Based on Temporal-Spatial Awareness and C-RBAC

In this chapter, aspects of the TA-RBAC-based  $m$ -IPS scheme that can detect mobile device security threats in business and social settings, including use of case scenarios, are discussed in detail. The components and constraints of  $m$ -IPS systems are outlined as follows.

(i) *Components:*

- (1) *user:* the person with authority to check time, locations, and roles  
 $U = \{\text{user1, user2, } \dots, \text{user } N\}$   
*role:* the specific work/tasks assigned to individual members  
 $R = \{\text{role1, role2, } \dots, \text{role } N\}$
- (2) *authority:* the permissions allocated to the user, consisting of time ( $t$ ) and location ( $L$ ) values  
 $P = \{\text{perm1, perm2, } \dots, \text{perm } N\}$   
 $\text{perm} = (t, \text{location})$
- (3) *time:* one of the conditions that constitute authority.  $T$  values consist of start values ( $ST$ ),

end values ( $ET$ ), and repeating cycles ( $C$ ). The cycles are divided into days, weeks, and months

$$T = \{t_1, t_2, \dots, t_n\}$$

$$t = (ST, ET, C)$$

$$ST, ET = (\text{year, month, day, hour, minute})$$

$$\text{Cycle} = (\text{day, week, month});$$

- (4) *location*: is one of the conditions that constitute authority.  $L$  values mean permitted places and consist of floor and room values

$$L \subseteq F \times R$$

$$L = \{\text{location}_1, \text{location}_2, \dots, \text{location}_N\}$$

$$\text{location} = (\text{floor, room})$$

$$F = \{\text{floor}_1, \text{floor}_2, \dots, \text{floor}_N\}$$

$$R = \{\text{room}_1, \text{room}_2, \dots, \text{room}_N\}.$$

(ii) *Constraints*:

- (1) *user-role*:  $UR \subseteq U \times R$

$$UR = (ur_1, ur_2, \dots, ur_N)$$

$$ur = (\text{user, role})$$

- (2) *role-permission*:  $RP \subseteq R \times P$

$$RP = (rp_1, rp_2, \dots, rp_N)$$

$$rp = (\text{role, perm})$$

$$rp = (\text{role, perm } (T (ST (\text{year, month, day, hour, minute}), ET (\text{year, month, day, hour, minute}), C (\text{day, week, month}))), \text{location } (\text{floor, room}))).$$

**4.1. Contextual Role-Based Access Control.** In RBAC, the user-role relationship is more dynamic than the role-permission relationship. As a result, context can be categorized into static constraints, for example, user nationality, salary, and so on, and dynamic constraints, for example, time, location, and purpose, of the user for which access request has been made. One approach to enforce the dynamic context oriented policies is to rapidly change the permission assignment relations that depend on the dynamic contexts. Another approach is to define permissions that should consider the static and dynamic behavior of context constraints. Based on this, the adoption of the existing well-known access control models and technologies is sensible as it provides a means to extend from traditional to context-based access control policies and facilitates obligation policies enforcement [2, 3, 12].

Mobile computing environments are characterized by many aspects, one of which is their potential size. Several definitions of the concept domain have been given in the literature.

**Definition 1** (domain). Domain is a logical bound defined over some space that contains at least one mobile device object, whereas space and mobile device object are identifiable by the mobile computing environment.

**Definition 2** (temporal domain). Temporal domain describes a logical bound that surrounds at least one or a list of mobile device objects and contains temporal roles identified by the mobile computing environment.

**Definition 3** (spatial domain). Spatial domain describes a logical bound that surrounds at least one or a list of mobile device objects and contains spatial roles identified by the mobile computing environment.

In general, RBAC is a very useful access control model but due to the distributed and heterogeneous nature of organizations, subject centric (traditional RBAC) is not sufficient. With the rapid advancement in technologies today, organizational resources are widely distributed in mobile computing environments. Also users can send request to access the resources at any time from any location. Under these circumstances, an extension of RBAC model is necessary in order to properly manage the organizational resources in multidomain environment keeping in mind the confidentiality, integrity, and availability. We used the C-RBAC model [12], an extension of traditional role-based access control model that allows security administrators to define context oriented access control policies enriched with the notion of purposes. By adding purpose roles, we extend traditional access control model that helps organizations to know *which* user can perform *what* operation on *which* object with *what* purpose. In this paper, we used the following CRBAC core elements [12, 16]: (1) user, roles, permissions, and user-to-role mappings; (2) mobile device objects which are the set of all mobile device objects; (3) role-to-permission mappings with the same meaning as permission-to-role mappings in this model; (4) the set of all entities related to the authorizations; (5) the set of all mappings; and finally (6) the set of sessions as tuple of  $\langle \text{user, role, permission} \rangle$ . Figure 1 should be used as the basics of extended C-RBAC model in this paper.

**4.2. System Architecture.** The existing wireless IPSs use pattern-based detection engines to determine whether to block mobile devices. They then record the results in ACLs and DB and provide functions to block or allow entirely. However, this method is vulnerable to attacks by individuals purporting to be permitted users, who can then easily act without any restriction on time, space, or roles. Furthermore, the existing IPS methods are unable to make allowances for exceptional circumstances with respect to time, spaces, or roles that may occur with mobile devices used in work and social settings. An *m*-IPS scheme is proposed in this study to address these difficulties in control and wireless security threats. The scheme determines the locations of mobile devices based on wireless signals picked up by sensors and first checks whether the locations and current time are allowed values. The profiles of the mobile devices accessed in mobile environments are then compared with stored profiles. Finally, the permissions allocated to the mobile devices are checked to ensure precise and safe access control of individual mobile devices. The *m*-IPS scheme is largely composed of *m*-IPS-ME agent, *m*-IPS-ME AP, *m*-IPS-ME sensors, *m*-IPS-ME server, and *m*-IPS-ME DB. Figure 2 shows the architecture of the *m*-IPS for the mobile environment with TA-RBAC.

The ***m*-IPS-ME agent** stores and manages the profiles of user devices. The profiles are used to check permissions when devices access networks. In addition, the *m*-IPS-ME

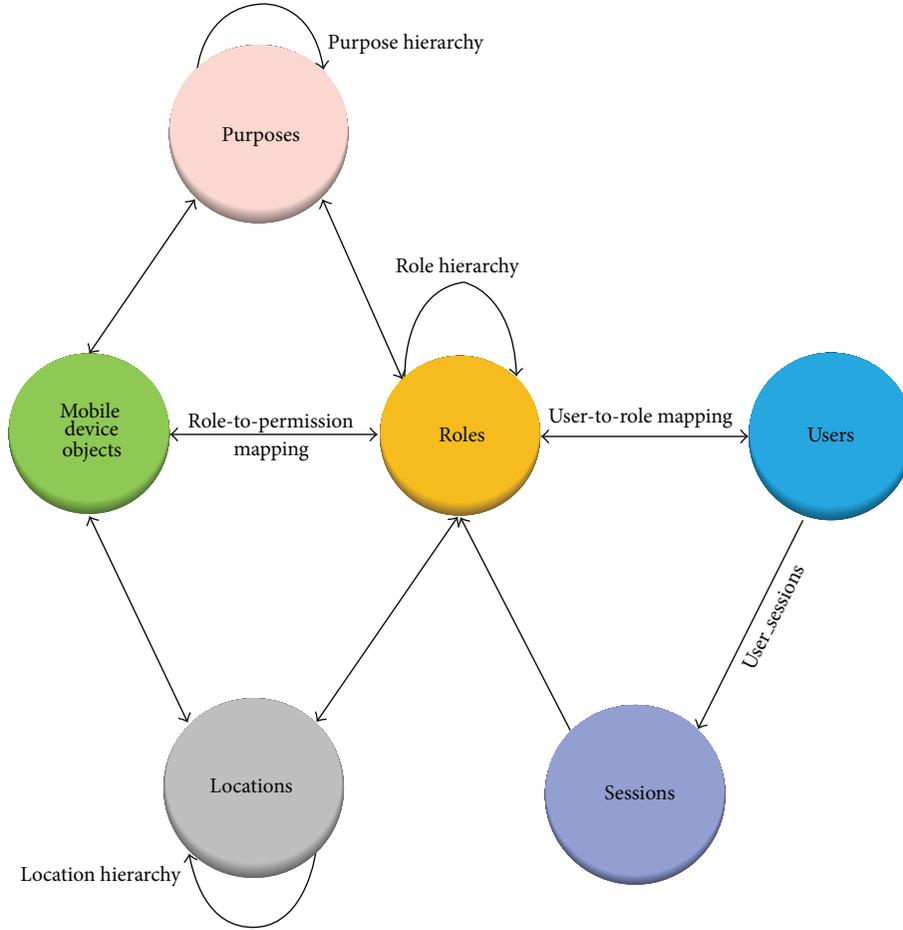


FIGURE 1: Contextual role-based access control model.

model features an access module for controlling the agent. The *m*-IPS-ME AP performs the communication function for the agent to access the wireless network. The *m*-IPS-ME sensor detects the location of the mobile device through the scanning module when the agent accesses the wireless network. The sensor management module controls the agent's communication based on the information sent from the server. The *m*-IPS-ME server registers the device profiles in advance. When the agent requests access to the network, the server compares the profile and the role information of the agent with the role information stored in the *m*-IPS-ME DB to allow communication by relevant devices to the sensor AP in case the two sets of information are identical to each other.

**4.3. Service Scenario.** The service scenario of the proposed *m*-IPS TA-RBAC system for the mobile environment is reviewed in this section. The coefficients employed in the service scenario are defined in Table 3.

Figure 3 illustrates the service scenario of the model. When the user wishes to access the wireless network in an office, meeting room, or social environment, the model checks whether the user's location and the current time are within the allowed ranges. Thereafter, the model compares the profile requested by the agent with the profile information

TABLE 3: Term and explanation.

| Term                    | Explanation                                             |
|-------------------------|---------------------------------------------------------|
| <i>m</i> -IPS-ME Agnt   | <i>m</i> -IPS mobile environments with TA-RBAC agent    |
| <i>m</i> -IPS-ME AP     | <i>m</i> -IPS mobile environments with TA-RBAC AP       |
| <i>m</i> -IPS-ME Sensor | <i>m</i> -IPS mobile environments with TA-RBAC sensor   |
| <i>m</i> -IPS-ME Svr    | <i>m</i> -IPS mobile environments with TA-RBAC server   |
| <i>m</i> -IPS-ME DB     | <i>m</i> -IPS mobile environments with TA-RBAC database |

stored in the database to see if they are the same. The agent role information should also match.

The detailed operation processes based on service scenario are as follows:

- (1) *m*-IPS-ME Agnt → *m*-IPS-ME AP: Req\_Conn (profile\_agnt)
- m*-IPS-ME AP → *m*-IPS-ME Svr: Req\_Conn (profile\_agnt)

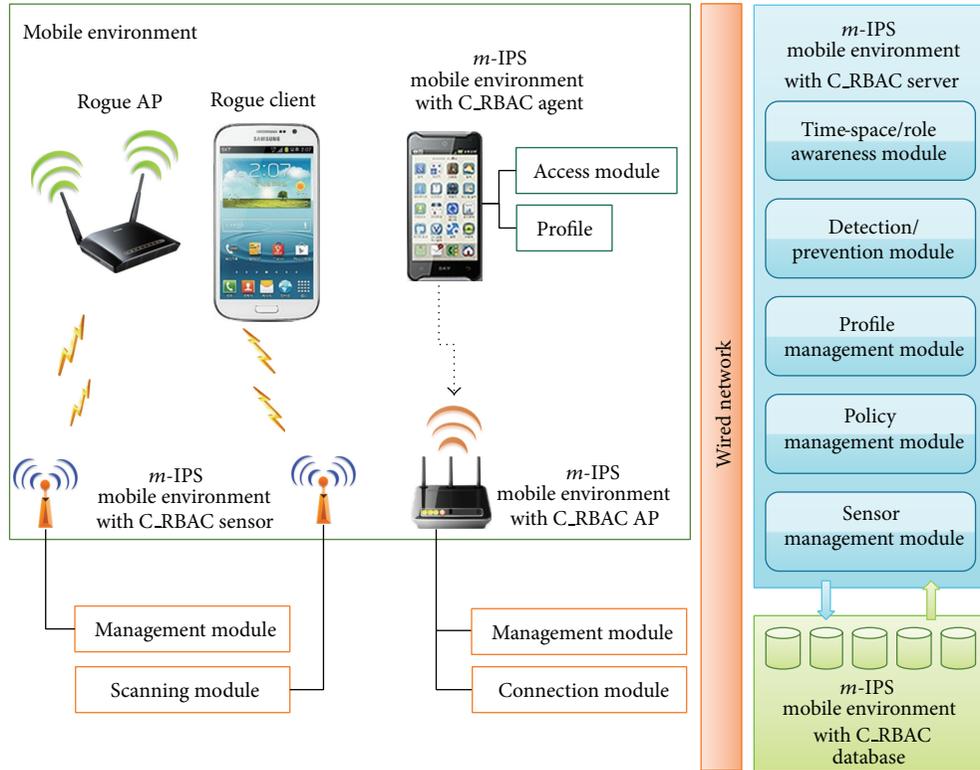


FIGURE 2: *m*-IPS for mobile environment with temporal-spatial awareness-RBAC (TA-RBAC) architecture.

- The *m*-IPS-ME Agnt sends the profile\_agnt information to the *m*-IPS-ME Svr through *m*-IPS-ME AP and requests a connection;
- (2) *m*-IPS-ME Svr  $\rightarrow$  *m*-IPS-ME DB: Req\_Profile  
*m*-IPS-ME DB  $\rightarrow$  *m*-IPS-ME Svr: Resp\_Profile (profile\_db)  
 To check the user's profile, the *m*-IPS-ME Svr sends a request to the *m*-IPS-ME DB for the profile of the relevant user to receive profile\_db;
  - (3) *m*-IPS-ME Svr  $\rightarrow$  *m*-IPS-ME sensor: Req\_Positioning  
 To identify the location of the *m*-IPS-ME Agnt, the *m*-IPS-ME Svr sends a request to the *m*-IPS-ME sensors to measure the signals;
  - (4) *m*-IPS-ME sensor: Scanning  
 Multiple *m*-IPS-ME sensors measure the intensity of the signals from the *m*-IPS-ME Agnt;
  - (5) *m*-IPS-ME sensor  $\rightarrow$  *m*-IPS-ME Svr: Resp\_Position (sig)  
 Multiple *m*-IPS-ME sensors transmit the information on the intensity of the signals from the *m*-IPS-ME Agnt to the *m*-IPS-ME Svr;
  - (6) *m*-IPS-ME Svr: Positioning()  
 It analyzes *m*-IPS-ME sensor signals to determine the location;
  - (7) *m*-IPS-ME Svr  $\rightarrow$  *m*-IPS-ME DB: Req\_Allowed List(pos,time)  
*m*-IPS-ME DB  $\rightarrow$  *m*-IPS-ME Svr: Resp\_Allowed List(pos,time)  
 The *m*-IPS-ME Svr checks whether the location of the *m*-IPS-ME Agnt and the current time are within the allowed ranges according to the *m*-IPS-ME DB;
  - (8) *m*-IPS-ME Svr: Decision()  
*m*-IPS-ME Svr: Compare(profile\_agnt, profile\_db)  
 After checking whether the location and the current time are within the allowed ranges based on the identified time and location information, the *m*-IPS-ME Svr judges whether to implement the second stage of authentication. In addition, the *m*-IPS-ME Svr compares the profile of the agent collected as set forth under (2) to the profile in the DB to judge whether to implement the third stage of authentication;
  - (9) *m*-IPS-ME Svr  $\rightarrow$  *m*-IPS-ME DB: Req\_AllowedList (role)  
*m*-IPS-ME DB  $\rightarrow$  *m*-IPS-ME Svr: Resp\_AllowedList (role)  
 The *m*-IPS-ME Svr checks whether the role of the agent is identical to the DB role information in the *m*-IPS-ME DB;
  - (10) *m*-IPS-ME Svr  $\rightarrow$  *m*-IPS-ME sensor: Req\_sensor\_control

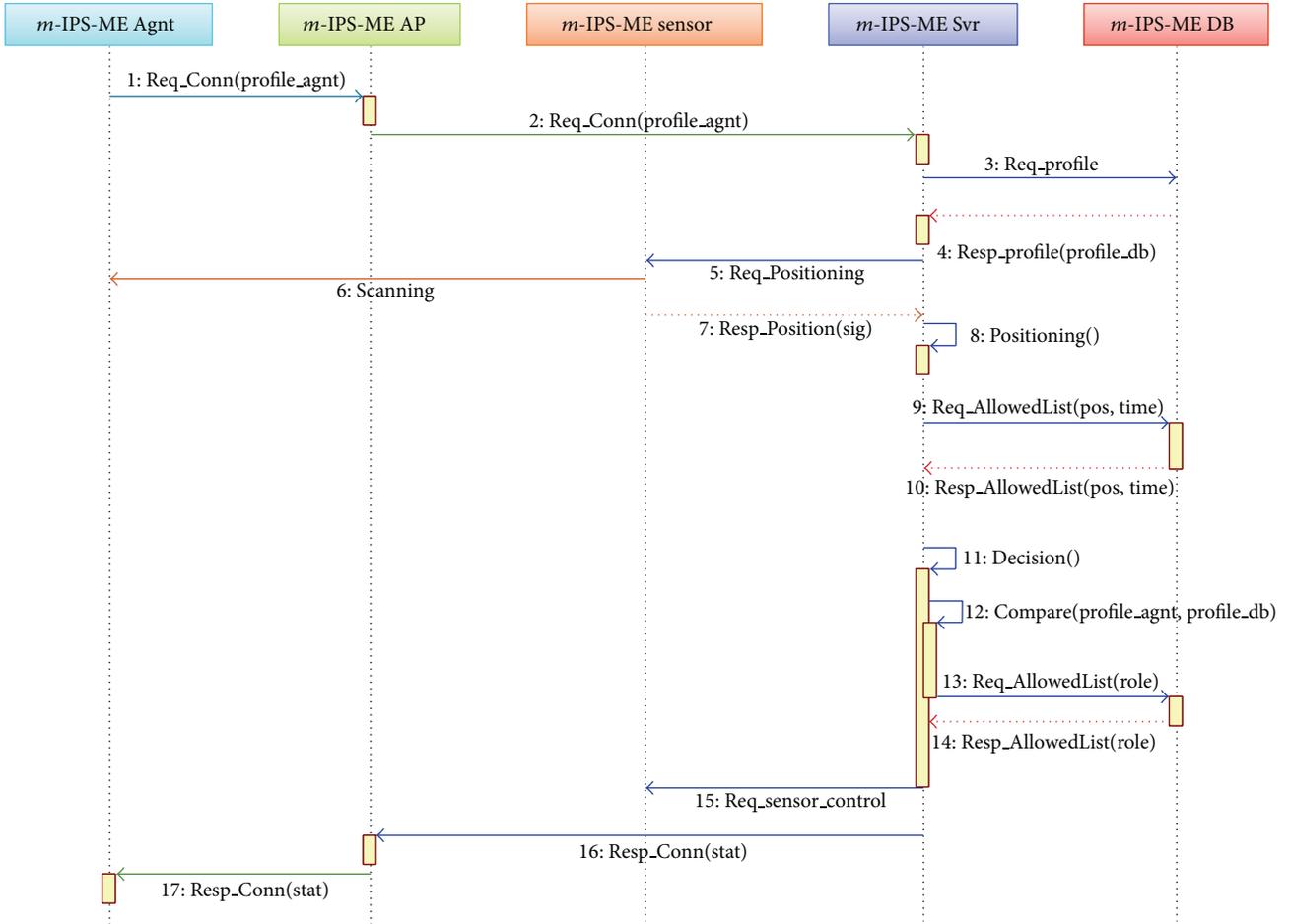


FIGURE 3: Service scenario.

Depending on whether the connection is allowed, the *m-IPS-ME Svr* requests the *m-IPS-ME Sensor* to restrict or not signals from the *m-IPS-ME Agnt*;

(11) *m-IPS-ME Svr* → *m-IPS-ME AP*: Resp\_Conn(stat)

*m-IPS-ME AP* → *m-IPS-ME Agnt*: Resp\_Conn(stat)

The *m-IPS-ME Svr* transmits connection information(stat) to the *m-IPS-ME Agnt* through the **wireless IPS-MO-AP**.

**4.4. Evaluation for Efficiency and Security.** In Table 4, the existing methods described in Section 2 and the proposed method are compared and analyzed. The methods were judged based on whether they can prevent wireless security threats that may occur in business/work and social settings utilizing mobile devices as set forth in Section 3. The method proposed by Wen-chu Hsieh focused only on detection by wireless IDSs and thus requires additional systems for prevention. The method proposed by Chen et al. improved false-positive rates using signature detection and planned recognition-based wireless IPSs but did not consider information on temporal-spatial elements and roles. Thus, false-positive rates are still not improved in mobile business and social environments where flexible access control is

necessary. As shown in Table 4, the performance of the *m-IPS* scheme described in this paper is superior to that of Sandhu et al. [3] and Nyanchara and Osborn [6]. The notation  $\bigcirc$  means the strong secure mechanism for providing to mobile environments, and  $\Delta$  means the medium secure method for mobile devices of mobile business, and finally  $\times$  is the weak point for security threats in mobile devices.

## 5. Conclusion

The use of diverse wireless devices, such as smartphones and smart pads, has increased rapidly in a short period. Work environments have also changed, with wired and wireless networks coexisting. Wireless IPSs are used to provide secure communication in these environments. However, the existing wireless IPSs are universal security systems equipped to deal only with general security. They have many problems due to the absence of temporal-spatial and role elements, and they are ill equipped to deal with security associated with wired/wireless composite work environments and offices. In the future, security threats in work environments are expected to become more frequent and to cause more damage.

TABLE 4: Characteristics comparison and analysis among the existing methods and the proposed method.

| Criteria                                     | Method            |                          |               |
|----------------------------------------------|-------------------|--------------------------|---------------|
|                                              | Wen-chu Hsieh [3] | Nyanchama and Osborn [6] | <i>m</i> -IPS |
| Rogue AP                                     | △                 | ○                        | ○             |
| Evil twin                                    | △                 | ○                        | ○             |
| MAC spoofing                                 | △                 | ○                        | ○             |
| MIMT                                         | △                 | ○                        | ○             |
| DoS attack                                   | △                 | ○                        | ○             |
| Honeypot                                     | △                 | ○                        | ○             |
| Access control with <i>m</i> -IPS and C_RBAC | ×                 | ×                        | ○             |

○: strong, △: medium, and ×: weak.

In this paper, the concept and the configuration of a wireless IPS were discussed, in addition to security threats and requirements in mobile environments by using mobile devices. Therefore, an efficient and secure mobile-IPS (*m*-IPS) has been proposed for businesses utilizing mobile devices in mobile environments for human-centric computing. This system incorporates temporal-spatial awareness and checks users' temporal-spatial information, profiles, and role information to provide precise access control. This research is meaningful in that access control is provided by checking users' temporal-spatial information, profiles, and role information, thereby leading to safer use of mobile devices in offices. To further improve the security of mobile devices, additional studies on the access modules used with these devices are necessary.

## Conflict of Interests

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

## Acknowledgment

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the CPRC (Communications Policy Research Center) support program supervised by the KCA (Korea Communications Agency) (KCA-2013-003).

## References

- [1] G. Chen, H. Yao, and Z. Wang, "An intelligent WLAN intrusion prevention system based on signature detection and plan recognition," in *Proceedings of the 2nd International Conference on Future Networks (ICFN '10)*, pp. 168–172, January 2010.
- [2] E. Georgakakis, S. A. Nikolidakis, D. D. Vergados, and C. Douligaris, "Spatio temporal emergency role based access control (STEM-RBAC): a time and location aware role based access control model with a break the glass mechanism," in *Proceedings of the 16th IEEE Symposium on Computers and Communications (ISCC '11)*, pp. 764–770, July 2011.
- [3] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Computer role-based access control models," *IEEE Computer Society*, vol. 29, no. 2, pp. 38–47, 1996.
- [4] X. Zhou, Y. Ge, X. Chen, Y. Jing, and W. Sun, "A distributed cache based reliable service execution and recovery approach in MANETs," *Journal of Convergence*, vol. 3, no. 1, pp. 5–12, 2012.
- [5] A. U. Bandaranayake, V. Pandit, and D. P. Agrawal, "Indoor link quality comparison of IEEE 802.11a channels in a multi-radio Mesh network testbed," *Journal of Information Processing Systems*, vol. 8, no. 1, pp. 1–20, 2012.
- [6] M. Nyanchama and S. Osborn, "The role graph model and conflict of interest," *ACM Transactions on Information and System Security*, vol. 2, no. 1, pp. 3–33, 1999.
- [7] S. Silas, K. Ezra, and E. B. Rajsingh, "A novel fault tolerant service selection framework for pervasive computing," *Human-Centric Computing and Information Sciences*, vol. 2, no. 5, pp. 1–14, 2012.
- [8] M. S. Kirkpatrick, G. Ghinita, and E. Bertino, "Privacy-preserving enforcement of spatially aware RBAC," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 5, pp. 627–640, 2012.
- [9] J. Timofte, "Wireless intrusion prevention system," *Revista Informatica Economica*, vol. 47, pp. 129–132, 2008.
- [10] Y. Zhang, G. Chen, W. Weng, and Z. Wang, "An overview of wireless intrusion prevention systems," in *Proceedings of the 2nd International Conference on Communication Systems, Networks and Applications (ICCSNA '10)*, pp. 147–150, July 2010.
- [11] W. Hsieh, C. Lo, J. Lee, and L. Huang, "The implementation of a proactive wireless intrusion detection system," in *Proceedings of the 4th International Conference on Computer and Information Technology (CIT '04)*, pp. 581–586, IEEE Press, September 2004.
- [12] M. N. Tahir, "C-RBAC: contextual role-based access control model," *Ubiquitous Computing and Communication Journal*, vol. 2, no. 3, pp. 67–74, 2007.
- [13] D. Lijun, Y. Shengsheng, X. Tao, and L. Rongtao, "WBIPS: a lightweight WTLS-based intrusion prevention scheme," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2298–2301, IEEE Press, September 2007.
- [14] A. Vartak, S. Ahmad, and K. N. Gopinath, "An experimental evaluation of Over-The-Air (OTA) wireless intrusion prevention techniques," in *Proceedings of the 2nd International Conference on Communication Systems Software and Middleware*, pp. 1–7, IEEE Computer Society, January 2007.
- [15] G. Chen, H. Yao, and Z. Wang, "Research of wireless intrusion prevention systems based on plan recognition and honeypot," in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '09)*, pp. 1–5, IEEE Computer Society, November 2009.
- [16] D. Zou, L. He, H. Jin, and X. Chen, "CRBAC: imposing multi-grained constraints on the RBAC model in the multi-application environment," *Journal of Network and Computer Applications*, vol. 32, no. 2, pp. 402–411, 2009.

## Research Article

# Exponential Stability for Impulsive Stochastic Nonlinear Network Systems with Time Delay

Lanping Chen,<sup>1,2</sup> Zhengzhi Han,<sup>1</sup> and Zhenghua Ma<sup>2</sup>

<sup>1</sup> School of Electronic, Information and Electrical Engineering, Shanghai Jiaotong University, Shanghai 200240, China

<sup>2</sup> College of Information and Engineering Science, Changzhou University, Jiangsu 213164, China

Correspondence should be addressed to Lanping Chen; lanping.chen@gmail.com

Received 4 December 2013; Revised 29 January 2014; Accepted 6 February 2014; Published 17 March 2014

Academic Editor: Laurence T. Yang

Copyright © 2014 Lanping 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.

We study the exponential stability of the complex dynamical network described by differentially nonlinear equations which couple with time delay and stochastic impulses. Some sufficient conditions are established to ensure  $p$ th moment exponential stable for the stochastic impulsive systems (SIS) with time delay. An example with its numerical simulation is presented to illustrate the validation of main results.

## 1. Introduction

As the extension and expansion of Internet network, the Internet of things is the complex networks which are made up of interconnected nodes and used to describe various systems of real world. In many systems such as signal processing systems, computer networks, automatic control systems, flying object motions, and telecommunications, impulsive effects are common phenomena due to instantaneous perturbations at certain moments. Therefore, the study of the dynamical networks with impulsive effects is important for understanding the dynamical behaviors of the most real-world complex networks. The impulsive dynamic systems have been studied extensively (see [1–4] and references therein). In addition to impulsive effects, stochastic effects likewise exist in real systems. In recent years stochastic impulsive dynamic system is an emerging field drawing attention from various disciplines of sciences and engineering.

Many real-world problems in science and engineering can be modeled by nonlinear stochastic impulsive dynamic systems (see [5, 6] and references therein). The stability analysis is much more complicated because of the existence of simultaneous impulsive effects and stochastic effects. So far, there are several results on impulsive stochastic systems,

which we can find in [7–10]. However, to the best of the authors' knowledge, little study on impulsive stabilization of stochastic delay systems has been done so far. Motivated by the above consideration, in this paper we analysis this system and obtain sufficient conditions to ensure the  $p$ th moment asymptotic stability of stochastic impulsive systems with arbitrarily infinite delays. It is shown that an unstable stochastic delay system can be successfully stabilized by impulses and the results can be easily applied to stochastic systems with arbitrarily time delays.

## 2. Preliminaries

Let  $R^n$  denote the  $n$ -dimensional real space and let  $\tau > 0$  be a positive real number. Let  $PC([-\tau, 0]; R^n)$  denotes the family of piecewise continuous functions from  $[-\tau, 0]$  to  $R^n$ .  $PC([-\tau, 0]; R^n) = \varphi : [-\tau, 0] \rightarrow R^n | \varphi(t^+) = \varphi(t), \varphi(t^-) = \varphi(t)$  for  $t \in (-\tau, 0]$ , with the norm  $\|\varphi\| = \sup_{-\tau \leq \theta \leq 0} |\varphi(\theta)|$ , where  $\varphi(t^+)$  and  $\varphi(t^-)$  denote the right-hand and left-hand limit of function  $\varphi(t)$  at  $t$ , respectively.

Consider the impulsive stochastic differential equation as follows:

$$dx(t) = f(t, x(t), x_t) dt + g(t, x_t) dw(t), \quad t \geq 0, t \neq t_k,$$

$$\begin{aligned} \Delta x(t_k) &= x(t_k^+) - x(t_k) = I_k(x(t_k)), \\ t &= t_k, \quad k = 1, 2, \dots, m, \\ x(t_0) &= \xi, \quad t = [-\tau, 0], \end{aligned} \tag{1}$$

where the initial value  $\xi \in \text{PC}([-\tau, 0]; R^n)$ ,  $x(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T$ ,  $x_t$  is regarded as a PC-valued stochastic process,  $x_t = \{x(t + \theta) : -\tau \leq \theta \leq 0\}$ ,  $f : R^+ \times R^n \times \text{PC}([-\tau, 0]; R^n) \rightarrow R^n$ ,  $g : R^+ \times \text{PC}([-\tau, 0]; R^n) \rightarrow R^{n \times m}$ , and  $w(t)$  is an  $m$ -dimensional standard Brownian motion defined on the complete probability space.

*Definition 1.* Let  $C^{2,1}(R^n \times [t_0 - \tau, \infty); R^+)$  denote the family of all nonnegative functions  $V(x, t)$  on  $R^n \times [t_0 - \tau, \infty)$  that are continuously twice differentiable in  $x$  and once in  $t$ . For a  $V \in C^{2,1}(R^n \times [t_0 - \tau, \infty); R^+)$ , one can define the Kolmogorov operator  $\mathcal{L}V$  as follows:

$$\begin{aligned} \mathcal{L}V(x, t) &= V_t(x, t) + V_x(x, t) f(x, t) \\ &\quad + \frac{1}{2} \text{Tr} \{g^T(x, t) V_{xx} g(x, t)\}, \end{aligned} \tag{2}$$

where  $V_t = \partial V(x, t) / \partial t$ ,  $V_x = (\partial V(x, t) / \partial x_1, \dots, \partial V(x, t) / \partial x_n)$ , and  $V_{xx} = \partial^2 V(x, t) / \partial x^2$ .

*Definition 2.* The trivial solution of SIS (1) is said to be the  $p$ th moment exponential stable if there exist positive constants  $\alpha > 0$  and  $K \geq 1$  such that

$$E\|x(t)\|^p \leq K e^{-\alpha(t-t_0)} \|x_0\|^p, \quad t > t_0, \quad t \in R^+. \tag{3}$$

The following lemmas can be found in [11].

**Lemma 3.** Let  $x, y \geq 0$ ,  $a, b > 1$ , then

$$xy \leq \frac{x^a}{a} + \frac{y^b}{b}, \quad \frac{1}{a} + \frac{1}{b} = 1. \tag{4}$$

**Lemma 4.** Let  $x, y \geq 0$ ,  $p \geq j \geq 0$ , then

$$x^{p-j} y^j \leq \frac{(p-j)x^p + jy^p}{p}. \tag{5}$$

### 3. Main Results

In this section, we shall focus on sufficient conditions to achieve exponential stability of the SIS by employing Razumikhin techniques and Lyapunov functions. Moreover, we will design the impulsive control for the stabilization of unstable stochastic systems by using the obtained results.

**Theorem 5.** If there exist positive constants  $p, c_1, c_2, \lambda, d_k > 1$ , and suppose there exists a function  $V$  such that

- (i)  $c_1|x|^p \leq V(x, t) \leq c_2|x|^p$ ;
- (ii)  $EV(x, t_k^+) \leq d_k EV(x, t_k)$ ;

(iii)  $E\mathcal{L}V(\varphi, t) \leq cEV(\varphi, t)$ , for all  $t \in (t_{k-1}, t_k]$ ;

(iv)  $\ln d_k \leq \lambda(t_k - t_{k-1})$ ,  $k = 1, 2, \dots$

Then the corresponding system (1) is the  $p$ th moment exponential stable.

*Proof.* For any  $t \in [t_1, t_2]$ , we can get from the conditions (ii) and (iii)

$$\begin{aligned} EV(t) &= EV(t_1^+) + \int_{t_1}^t cEV(x(s), s) ds \\ &\leq d_1 \left[ EV(0) + \int_0^{t_1} cEV(x(s), s) ds \right] \\ &\quad + \int_{t_1}^t cEV(x(s), s) ds = d_1 EV(0) \\ &\quad + d_1 \int_0^{t_1} cEV(x(s), s) ds + \int_{t_1}^t cEV(x(s), s) ds. \end{aligned} \tag{6}$$

In general for  $t \in [t_{k-1}, t_k]$ , one can find that

$$EV(t) \leq \prod_{0 \leq t_k \leq t} d_k EV(0) + \int_0^t \prod_{s \leq t_k \leq t} d_k cEV(x(s), s) ds. \tag{7}$$

From condition (iv), we get

$$\begin{aligned} \prod_{s \leq t_k \leq t} d_k &\leq e^{\lambda(t_2-t_1)} \cdot e^{\lambda(t_3-t_2)} \dots e^{\lambda(t_k-t_{k-1})} \\ &= e^{\lambda(t_k-t_1)} = e^{\lambda(t-s)} \cdot e^{\lambda(t_k-t)} \cdot e^{\lambda(s-t_1)}. \end{aligned} \tag{8}$$

For  $t \in [t_{k-1}, t_k], t_1, t_2, \dots, t_k$  be impulsive points in  $[s, t], t > s, \lambda < 0$ , then we obtain

$$\prod_{s \leq t_k \leq t} d_k \leq e^{\lambda(t-s)} \cdot e^{\lambda(t_k-t)} \leq e^{\lambda(t-s)} \cdot e^{\lambda(t_k-t_{k-1})} \leq \gamma e^{\lambda(t-s)}. \tag{9}$$

By (7) and (8), then we can get

$$\begin{aligned} EV(t) &\leq \gamma EV(0) e^{\lambda t} + \int_0^t \gamma e^{\lambda(t-s)} cEV(x(s), s) ds \\ &\leq \gamma \sup_{-\tau \leq \theta \leq 0} EV(\sigma) e^{\lambda t}. \end{aligned} \tag{10}$$

It follows from condition (i), that

$$c_1 E|x(t)|^p \leq EV(t) \leq \gamma \sup_{-\tau \leq \theta \leq 0} EV(\sigma) e^{\lambda t} \leq \gamma c_2 E\|\xi\|^p e^{\lambda t}, \tag{11}$$

which implies

$$E|x(t)|^p \leq \frac{\gamma c_2}{c_1} E\|\xi\|^p e^{\lambda t}. \tag{12}$$

System (1) is the  $p$ th moment exponentially stable. The proof is complete.  $\square$

**Theorem 6.** Assume that

- (i)  $EV_t \leq \eta EV(x(t)), EV_x \leq \bar{\eta}_1 EV(x(t))^{(p-1)/p}, EV_{xx} \leq \bar{\eta}_2 EV(x(t))^{(p-2)/p};$
- (ii)  $E|f(x_t)|^p \leq \bar{\eta}_3 \sup_{-\tau \leq \theta \leq 0} EV(x_t),$   
 $E(\text{Tr}\{g(x_t)^T g(x_t)\})^{p/2} \leq \bar{\eta}_4 \sup_{-\tau \leq \theta \leq 0} EV(x(t-\tau));$
- (iii)  $EV(x(t+\theta)) \leq qEV(x(t)), \theta \in (-\tau, 0];$

and conditions of Theorem 5 hold simultaneously, then the system is  $p$ th exponential stable, where  $E$  denotes the expectation.

*Proof.* Take

$$\mathcal{L}V = V_t + V_x f(x_t) + \frac{1}{2} \text{Tr}\{g^T(x_t) V_{xx} g(x_t)\} \quad (13)$$

then take the mathematical expectation of both sides of the Formula (12), we obtain

$$E\mathcal{L}V = EV_t + E(V_x f) + \frac{1}{2} E[\text{Tr}\{g^T V_{xx} g\}]. \quad (14)$$

Using Lemma 3, from (i) and (ii), we obtain

$$\begin{aligned} & E(V_x f(x, x_t)) \\ & \leq EV_x \cdot E f(x, x_t) \leq \bar{\eta}_1 E(V)^{(p-1)/p} \cdot E|f(x, x_t)| \\ & \leq \frac{\bar{\eta}_1(p-1)}{p} EV + \frac{1}{p} \bar{\eta}_3 \cdot E|f(x, x_t)|^p \leq \frac{\bar{\eta}_1(p-1)}{p} EV \\ & \quad + \frac{1}{p} \bar{\eta}_3 \cdot q \cdot \sup_{-\tau \leq \theta \leq 0} EV(x(t)), \end{aligned}$$

$$\begin{aligned} & E[\text{Tr}\{g^T(x_t) V_{xx} g(x_t)\}] \\ & \leq EV_{xx} \cdot E[\text{Tr}\{g^T(x_t) g(x_t)\}] \leq \bar{\eta}_2 E(V(x))^{(p-2)/p} \\ & \quad \cdot E(\text{Tr}\{g^T(x_t) g(x_t)\}) \leq \frac{\bar{\eta}_2(p-2)}{p} EV(x) \\ & \quad + \frac{2}{p} E[\text{Tr}\{g^T g\}]^{p/2} \leq \frac{\bar{\eta}_2(p-2)}{p} EV(x) \\ & \quad + \frac{2\bar{\eta}_4}{p} \cdot \sup_{-\tau \leq \theta \leq 0} EV(x(t-\tau)) \leq \frac{\bar{\eta}_2(p-2)}{p} EV(x) \\ & \quad + \frac{2\bar{\eta}_4}{p} \cdot q \cdot \sup_{-\tau \leq \theta \leq 0} EV(x(t)). \end{aligned} \quad (15)$$

It follows that

$$\begin{aligned} E\mathcal{L}V & \leq \bar{\eta}_1 EV(x(t)) + \frac{\bar{\eta}_1(p-1)}{p} EV(x(t)) \\ & \quad + \frac{\bar{\eta}_2(p-2)}{p} EV(x) + \frac{1}{p} \bar{\eta}_3 \cdot q \cdot \sup_{-\tau \leq \theta \leq 0} EV(x(t)) \\ & \quad + \frac{2\bar{\eta}_4}{p} \cdot q \cdot \sup_{-\tau \leq \theta \leq 0} EV(x(t)). \end{aligned} \quad (16)$$

Consequently, by the above statement, the conditions of Theorem 5 are all satisfied. Then, the conclusion follows from Theorem 5 and the proof is complete.  $\square$

*Remark 7.* From the above consequence, we know that the unstable stochastic system  $dx(t) = f(t, x, x_t)dt + g(t, x_t)dw(t)$  can be exponentially stabilized by the impulsive control  $u_k(x) = I_k(x), t_k, k \in N$ . Moreover, the steps of the impulsive control design satisfy the conditions of Theorem 5.

*Remark 8.* Consider a special case of system (1) shown as follows:

$$dx(t) = [Ax + f_1(t, x(t-\tau))]dt + g(t, x(t-\tau))dw(t), \quad t \geq 0, \quad t \neq t_k,$$

$$\Delta x(t_k) = x(t_k^+) - x(t_k) = I_k(x(t_k)),$$

$$t = t_k, \quad k = 1, 2, \dots, m,$$

$$x(t_0) = \xi, \quad t = [-\tau, 0], \quad (17)$$

there exist nonnegative function  $\alpha_i(t), \beta_i(t)$  such that

$$(i) |f_1(t, x(t-\tau))| \leq \sum_{i=1}^m \alpha_i(t) |x(t-\tau_i)|,$$

$$(ii) \|g(t, x(t-\tau))\|_F^2 \leq \sum_{i=1}^m \beta_i(t) |x(t-\tau_i)|^2,$$

where  $\tau = \max_{1 \leq i \leq m} \tau_i, \lambda_{\max}(\cdot)$  denotes the largest eigenvalue of a symmetric matrix. Then we derive the following theorem.

**Theorem 9.** Assume that there exist positive constants  $\kappa_1, \kappa_2, \kappa_3$  such that

$$\begin{aligned} \kappa_1 + (p-1)\kappa_2 + \frac{(p-2)\kappa_3}{2} & > 0, \\ \kappa_2 + \kappa_3 & > 0 \end{aligned} \quad (18)$$

hold, where  $\kappa_1 = \lambda_{\max}(A), \kappa_2 = \Theta_{i=1}^m \alpha_i(t), \kappa_3 = (p-1) \Theta_{i=1}^m \beta_i(t)$ , and if the conditions of Theorems 5 and 6 are satisfied, then the trivial solution of system (17) is  $p$ -moment exponentially stable.

*Proof.* Let  $P = Q^T Q$ , and  $V(t, x) = (x^T P x)^{p/2} = |Qx|^p$ . Then by Itô formula, we have

$$\begin{aligned} \mathcal{L}V & = V_t + V_x [Ax + f_1(x(t-\tau))] + \frac{1}{2} \text{Tr}\{g^T V_{xx} g\} \\ & = p|Qx|^{p-1} [QAx + Qf_1(t, x(t-\tau))] + \frac{p(p-1)}{2} \\ & \quad \times |Qx|^{p-2} \text{Tr}\{g^T(t, x(t-\tau)) Pg(t, x(t-\tau))\}. \end{aligned} \quad (19)$$

By condition (ii), we have

$$\begin{aligned} & \text{Tr}\{g^T(t, x(t-\tau)) Pg(t, x(t-\tau))\} \\ & = \|Qg(t, x(t-\tau))\|_F^2 \leq \sum_{i=1}^m \beta_i(t) |Qx(t-\tau_i)|^2. \end{aligned} \quad (20)$$

Substituting (20) into (19), and using conditions, we obtain

$$\begin{aligned} \mathcal{L}V &\leq p|Qx|^{p-1} [\lambda_{\max}(A) |Qx| + \sum_{i=1}^m \alpha_i(t) |Qx(t - \tau_i)|] \\ &\quad + \frac{p(p-1)}{2} |Qx|^{p-2} \sum_{i=1}^m \beta_i(t) |Qx(t - \tau_i)|^2. \end{aligned} \tag{21}$$

Using Lemma 4, we get

$$\begin{aligned} \mathcal{L}V &\leq |Qx|^p \left[ \lambda_{\max}(A) + (p-1) \sum_{i=1}^m \alpha_i(t) \right. \\ &\quad \left. + \frac{(p-1)(p-2)}{2} \sum_{i=1}^m \beta_i(t) \right] \\ &\quad + [\sum_{i=1}^m \alpha_i(t) + (p-1) \sum_{i=1}^m \beta_i(t)] |Qx(t - \tau_i)|^p \\ &= \left[ \kappa_1 + (p-1) \kappa_2 + \frac{(p-2) \kappa_3}{2} \right] V(t, x) \\ &\quad + [\kappa_2 + \kappa_3] V(t, x(t - \tau_i)). \end{aligned} \tag{22}$$

Summing up the above statements, we can see that all the conditions of Theorem 5 and condition (iii) of Theorem 6 are satisfied. Then the conclusion follows from Theorem 6 immediately and the proof is completed.  $\square$

### 4. Example

In this section, we present an example to demonstrate our theoretical results. Considering a nonlinear stochastic impulsive system as follows:

$$\begin{aligned} dx(t) &= f(x(t))dt + g(x(t), x(t - \tau))dw(t), \\ t &\geq 0, \quad t \neq t_k, \end{aligned} \tag{23}$$

$$\Delta x(t_k) = I_k(x(t_k)), \quad t = t_k, \quad k \in N,$$

where  $f(x) = x(t)$ ,  $g(x, x_t) \leq (1/4)(x^2 + x_t^2)$ ,  $I_k = -0.4$ ,  $\tau = 2$ .

*Step 1.* Calculate the parameters.

Without loss of generality, we choose  $c_1 = c_2 = 1$ ,  $p = 2$ ,  $d_k = 0.37$  such that they satisfy the conditions of (i) and (ii) of Theorem 5.

*Step 2.* Choose  $V(x, t) = x^2$ , then it is easy to calculate from the Itô formula that

$$\begin{aligned} E\mathcal{L}V(x(t), x(t - \tau)) &= 2E|x(t)|^2 + E|g(x, x_t)|^2 \\ &\leq \frac{9}{4}EV(x(t)) + \frac{1}{4}EV(x(t - \tau)) \end{aligned} \tag{24}$$

which satisfies condition (iii) of Theorem 5

$$E\mathcal{L}V(x(t), x(t - \tau)) \leq cEV(x(t)), \tag{25}$$

where take  $q = 5$ ,  $\lambda = 0.5$ ,  $t_{k+1} - t_k = 0.2$ .

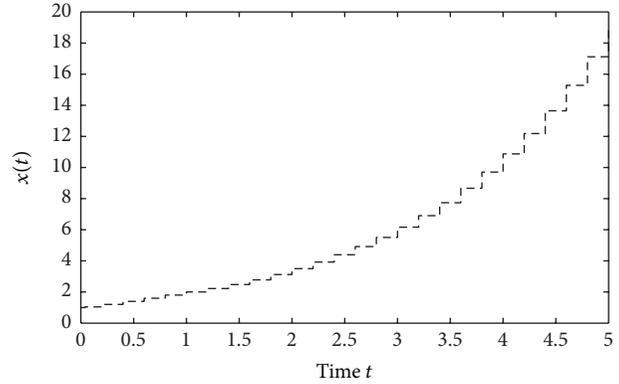


FIGURE 1: Instability of the stochastic delay system (23) without impulsive effect.

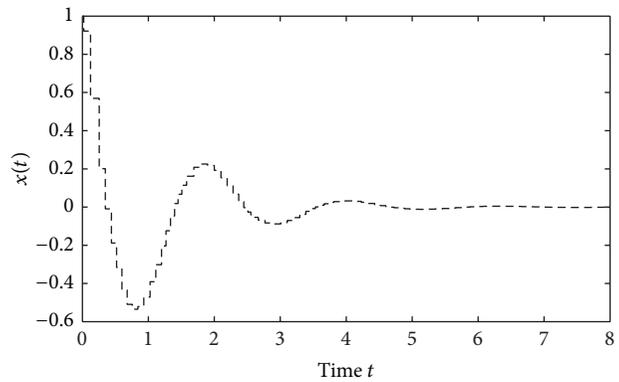


FIGURE 2: Impulsive Stabilization of the stochastic delay system (23).

*Step 3.* By calculation, we obtain  $c = 3.5$ , then

$$\ln d_k = -1.02 < -(c + \lambda)(t_{k+1} - t_k) = -0.8. \tag{26}$$

It satisfies condition (iv) of Theorem 5 which means that the system (23) is exponentially stable. Figure 1 gives the trajectory of the state of (23). It is obvious that the system is not stable without impulsive effect. Figure 1 shows that the solution of the stochastic delayed system (23) is unstable. Figure 2 shows the stability of the delay system with the impulsive controller.

### 5. Conclusion

In this paper, we have investigated the  $p$ -moment stability and applied the technique of Razumikhin techniques and Lyapunov functions to impulsive stochastic systems. Some sufficient conditions about the stability of impulsive stochastic systems in terms of two measures are derived. As a beneficial supplement in the study of impulsive stochastic systems with time delay, the concluded criteria are not only effective but also convenient in practical applications of specific systems in engineering and physics, etc. We also provided an illustrative example to show the effectiveness of our results.

## Conflict of Interests

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

## Acknowledgments

The authors are grateful for the support of the National Natural Science Foundation of China (Grant no. 61074003). This work is supported by the National Natural Science Foundation of China (no. 61074003).

## References

- [1] S. Zhang, J. Sun, and Y. Zhang, "Stability of impulsive stochastic differential equations in terms of two measures via perturbing Lyapunov functions," *Applied Mathematics and Computation*, vol. 218, no. 9, pp. 5181–5186, 2012.
- [2] Q. Wang and X. Liu, "Impulsive stabilization of delay differential systems via the Lyapunov-Razumikhin method," *Applied Mathematics Letters*, vol. 20, no. 8, pp. 839–845, 2007.
- [3] X. Song and A. Li, "Stability and boundedness criteria of nonlinear impulsive systems employing perturbing Lyapunov functions," *Applied Mathematics and Computation*, vol. 217, no. 24, pp. 10166–10174, 2011.
- [4] Y. Liu and S. Zhao, "A new approach to practical stability of impulsive functional differential equations in terms of two measures," *Journal of Computational and Applied Mathematics*, vol. 223, no. 1, pp. 449–458, 2009.
- [5] K. Liu, *Stability of Infinite Dimensional Stochastic Differential Equations with Applications*, vol. 135, Chapman & Hall/CRC, London, UK, 2006.
- [6] L. Wan and J. Duan, "Exponential stability of non-autonomous stochastic partial differential equations with finite memory," *Statistics & Probability Letters*, vol. 78, no. 5, pp. 490–498, 2008.
- [7] P. Cheng, F.-Q. Deng, and X.-S. Da, "Razumikhin-type theorems for asymptotic stability of impulsive stochastic functional differential systems," *Journal of Systems Science and Systems Engineering*, vol. 19, no. 1, pp. 72–84, 2010.
- [8] S. Peng and Y. Zhang, "Razumikhin-type theorems on  $p$ th moment exponential stability of impulsive stochastic delay differential equations," *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1917–1922, 2010.
- [9] B. Liu, "Stability of solutions for stochastic impulsive systems via comparison approach," *IEEE Transactions on Automatic Control*, vol. 53, no. 9, pp. 2128–2133, 2008.
- [10] X. Mao, G. G. Yin, and C. Yuan, "Stabilization and destabilization of hybrid systems of stochastic differential equations," *Automatica*, vol. 43, no. 2, pp. 264–273, 2007.
- [11] Q. Song and Z. Wang, "Stability analysis of impulsive stochastic Cohen-Grossberg neural networks with mixed time delays," *Physica A*, vol. 387, no. 13, pp. 3314–3326, 2008.

## Research Article

# Ubiquitous Health Management System with Watch-Type Monitoring Device for Dementia Patients

**Dongmin Shin, Dongil Shin, and Dongkyoo Shin**

*Department of Computer Engineering, Sejong University, 98 Gunja-Dong, Gwangjin-Gu, Seoul 143-747, Republic of Korea*

Correspondence should be addressed to Dongkyoo Shin; [shindk@sejong.ac.kr](mailto:shindk@sejong.ac.kr)

Received 11 November 2013; Revised 13 January 2014; Accepted 19 January 2014; Published 4 March 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Dongmin Shin 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.

For patients who have a senile mental disorder such as dementia, the quantity of exercise and amount of sunlight are an important clue for doses and treatment. Therefore, monitoring daily health information is necessary for patients' safety and health. A portable and wearable sensor device and server configuration for monitoring data are needed to provide these services for patients. A watch-type device (smart watch) that patients wear and a server system are developed in this paper. The smart watch developed includes a GPS, accelerometer, and illumination sensor, and can obtain real time health information by measuring the position of patients, quantity of exercise, and amount of sunlight. The server system includes the sensor data analysis algorithm and web server used by the doctor and protector to monitor the sensor data acquired from the smart watch. The proposed data analysis algorithm acquires the exercise information and detects the step count in patients' motion acquired from the acceleration sensor and verifies the three cases of fast pace, slow pace, and walking pace, showing 96% of the experimental results. If developed and the u-Healthcare System for dementia patients is applied, higher quality medical services can be provided to patients.

## 1. Introduction

The increase in the elderly population due to the development of medical technology is creating challenges for care professionals and developers of ubiquitous healthcare systems.

Dementia refers to the cognitive impairment usually affecting old people and makes functioning in daily life more difficult. Early symptoms of dementia include memory loss gradually affecting everyday activities. Typically from a few months to several years, the first symptoms are mild but develop slowly and gradually lead to serious memory loss. In addition, dementia patients have difficulty in recognizing their family members and doing complicated tasks. They usually have wandering symptoms and more than 73% experience being lost or missing [1].

The ubiquitous healthcare system is a convergence of information communication technology and healthcare and has emerged in various ways to help these kinds of patients [2]. Keruve, a Spanish company, provides a medical service for dementia patients. This service uses a bracelet with a built-in GPS and a portable device. The GPS bracelet features precise location detection using triangulation, even if the

patient is in the room [3]. Korea Telecom, a Korean company, has developed a location-tracking system using GPS and Code Division Multiple Access (CDMA) [4]. Gangnam District Office in Seoul, Korea, has developed a system called Gangnam U-Safe System [5]. This service began in May 2009 using Ubiquitous Sensor Network (USN) technology and GPS. This system provides a compact device featured with an emergency alarm service used for the safety of socially vulnerable individuals including children and those with intellectual disabilities.

Currently, healthcare systems for patients with dementia are focusing on location tracking using a Global Positioning System (GPS). For patients with mental disorders, momentum monitoring and medical service profiling can manage their risks and enhance their quality of life [6, 7]. In this paper, we develop an ubiquitous health management system for dementia patients to improve their health and safety following the concept Internet of Things (IoT) [8–10]. The system consists of a wrist watch-type device and a server system. The device includes a built-in GPS, ambient light sensor, and acceleration sensor and communicates with the server system. The server system functions include the

creation of a personal profile for patients and monitoring a patient's location and measuring the amount of sunlight illumination and walking step count to use as medical data. The system helps dementia patients avoid the risk of being missing or lost by wandering symptoms.

## 2. Related Works

Recently, the concept of Internet of Things (IoT) has been applied in ubiquitous healthcare systems and services [8–10]. IoT is a novel paradigm of technologies that interconnect everyday objects with each other through the Internet exploiting multiple wireless communication interfaces and advancements in computing devices [11]. With the spread of smart phones and tablets loaded with various sensors such as GPS and accelerometers, higher quality services are provided to the users by connection of the information on the Web and real world [12].

With the advent of IoT, research on numerous medical services for patients has been performed [9, 10]. Research on wireless networking technologies for developing a mobile healthcare environment has been carried out and it leads into the concept of mobile IoT (m-IoT), which is a new healthcare connectivity paradigm that interconnects IP-based communication technologies such as IPv6 over low power WPAN (6LoWPAN) with emerging 4G networks for future Internet-based healthcare services [9]. Typically, healthcare services are comprised of the sensors acquiring biosignals and the servers processing the huge amount of biodata generated from the sensors. Service platforms that interconnect cloud computing, distributed processing, and high speed data processing systems following the concept of IoT are being researched for efficient healthcare services [10].

Studies on human movement detection and behavioral patterns have been carried out in various ways for healthcare services. The motion recognition algorithm based on a motion-tree is developed using the acceleration features of a mobile phone [12]. The motion detection algorithm is one of the basic methods for detecting the number of walking steps [13, 14]. Human movements are distinguished by a pattern recognition algorithm and a way of extracting various motions are developed from basic motion patterns and feature vectors of humans. This function reads normal and abnormal movements, for example, sitting, standing, and falling down, as well as the number of steps [15–18].

Position tracking using GPS is one of the data for measuring the momentum as well as the current position of the patient in a healthcare system. Recently research on indoor position tracking methods using Wi-Fi or other positioning schemes are being carried out because it is impossible to get a GPS signal indoors [19, 20].

## 3. Development of a Ubiquitous Health Management System

The system consists of a watch-type monitoring device and server. The monitoring device includes a GPS, 3-axis accelerometer, and ambient light sensor. It is worn on

the patient's wrists and periodically transfers his activity information to the server derived from his location and amount of light illumination detecting sun exposure. Then care professionals and doctors can monitor the patient's health condition through the webpage delivered by the server. The server identifies the location through the patient's data transferred from the monitoring device and measures the patient's activity information through the step number detection algorithm and creates a profile about the patient's health information, together with the amount of light illumination to detect sun exposure.

*3.1. Development of the Watch-Type Monitoring Devices.* In the monitoring device, location-tracing functions using a GPS sensor can monitor the present location and migration route of the patient. The ambient light sensor measures the amount of sunlight illumination exposed to the device and records it. The 3-axis acceleration sensor records the value of the  $x$ -,  $y$ -, and  $z$ -axis coordinate values in real time. The server can get the number of patient's steps through the step detection algorithm.

The values of the sensors are obtained through the real time transfer of the data through Transmission Control Protocol/Internet Protocol (TCP/IP) communication on the CDMA network. After connection to the server through a Short Message Service (SMS) such as Server Open SMS and Transmission Close SMS for transfers, the values of the sensors exchange data with each other. At this moment, the transfer of the data by contacting the server is scheduled according to the regular cycle defined by the user. The server can inform the care professional or patient by alarm in the case of special events such as injection time and escape from patient's safety zone of patient.

The monitoring device is designed to be worn easily using the form factor of a wrist watch and because it is held in position by a clamp, it can prevent a patient from taking it off or losing it. Thus, if a demented patient experiences emergency or wandering symptoms, the problem can be quickly dealt with. The internal block diagram of the watch-type monitoring device proposed in this paper is shown in Figure 1.

*3.2. Development of the Health Management Server.* The server system is composed of the receiver module for receiving the transmitted data from the monitoring device, the health management module analyzing data, and the webpages performing management functions and patient monitoring, as shown in Figure 2.

First, the receiver module manages the watch's connection through the SMS receiver while waiting for the monitoring device's SMS. The receiver module with the Connection SMS receives the accumulated data saved in the monitoring device as the defined protocols after assigning a socket and a thread using TCP/IP communication.

The health management module generates the patient profile by analyzing the transferred data. It checks whether the user moves out of the scope of the designated safety zone or not using the GPS sensor data. And it converts the ambient

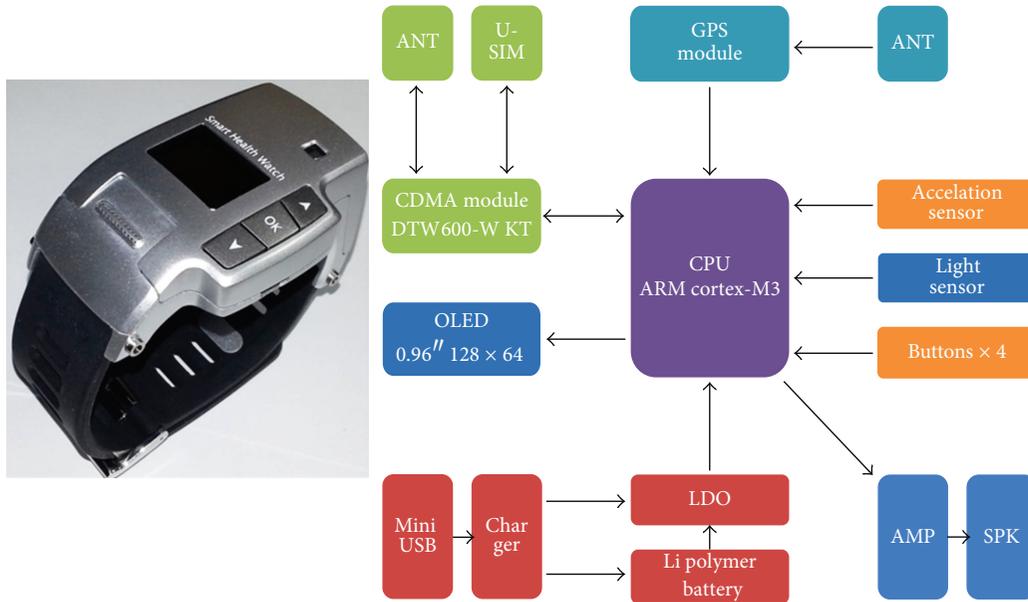


FIGURE 1: Watch-type monitoring device and its internal block diagram.

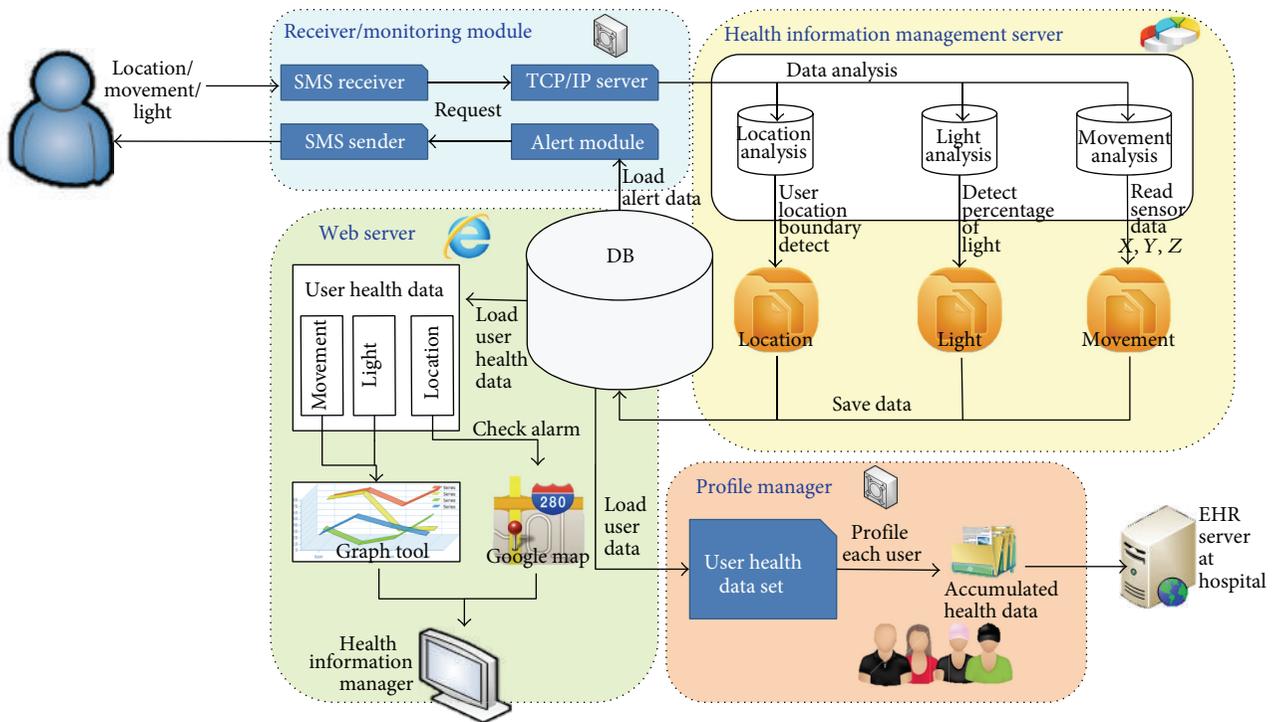


FIGURE 2: The System Operational Scenario.

light sensor data into a percentage from 0 to 100 accounting for the patient's exposure time to sunlight. Finally, it measures the amount of a patient's movement by counting walking steps based on the step detection algorithm using the 3-axis acceleration sensor data. The patient's data acquired from this module is separately saved into the database. The data in the database is used and recorded in the profile of each patient and can be monitored through the webpage.

The webpage is used to monitor the tracing location and health information of the patient obtained from the DB. First of all, a care professional can set up a communication period between the monitoring device and the server and the scope of the safety zone through the settings. The server indicates whether the traced patient's location is within the scope of the safety zone or not, and his present location and the scope of the safety zone would be marked in a circle on



FIGURE 3: Preprocessing of accelerometer data.

the map. The amount of sunlight indicates the exposure state hourly as the time-axis and exposure-axis through the graph. The activity mass also expresses the number of walk hourly through the graph. The health information can preserve the patient's health and safety because it monitors the patient's state through an activity list by time order, amount of sunlight, and location of the patient measured during outdoor activities.

#### 4. Walking Step Detection Algorithm

In addition to the location-tracking service for dementia patients, the system provides accurate walking step detection for use in healthcare. The step detection algorithm uses a 3-axis accelerometer to accurately detect a patient's steps and further analyzes his activities.

*4.1. Experimental Design.* The experiment done in this paper uses the watch-type monitoring device to compare the actual steps counted in 30~60 seconds with the value detected by the accelerometer under the same conditions. Eight people took part in this experiment creating 170 data of 3 types of steps—fast steps, normal steps, and slow steps every day. Each data is categorized in the database by experiment date, time, and the number of steps. Stored results are preprocessed into energy values for peak picking and analysis of distinctive features of the walk. Analyzed features are used to distinguish the step and nonstep activities and the measured number of steps is then compared to the actual number of steps counted.

*4.2. Preprocessing Data.* Figure 3 shows preprocessing of the accelerometer data. Each acquired  $x$ -,  $y$ -,  $z$ -axis data are in 8 byte double data types, recorded 80 times per second. It makes the calculation more efficient using the Signal Vector Magnitude (SVM) values than using 3 values simultaneously for each calculation. SVM in this experiment is expressed as the following equation (see Figure 4)

$$\text{SVM} = \sqrt{x_i^2 + y_i^2 + z_i^2}. \quad (1)$$

The accelerometer records 80 times per second and even catches subtle movements. Therefore, even if the patient is standing still, the accelerometer will be recording constantly changing values. These subtle noise signals could result in errors when measuring the number of steps. In this paper, we have used the Moving Average Filter (MAF) to filter out

these noises, preventing errors. The MAF has low pass filter properties and it can be expressed as follows:

$$\begin{aligned} T[n] &= \frac{1}{5} (\text{SVM}[n-2] + \text{SVM}[n-1] + \text{SVM}[n] \\ &\quad + \text{SVM}[n+1] + \text{SVM}[n+2]) \\ &= \frac{1}{5} \sum_{m=-2}^2 \text{SVM}[n-m]. \end{aligned} \quad (2)$$

Here, the value of  $n$ th MAF is denoted by  $T[n]$  and  $\text{SVM}[n-1]$  means  $(n-1)$ th SVM. Figure 5 shows the result of moving average filter.

*4.3. Step Detection Algorithm.* The step detection algorithm proposed in this paper finds the peaks from the preprocessed data and then counts the number of peak values that are over the threshold value, which is calculated from the data.

First, to pick out the peaks, we find the wave's mean gradient by computing the average of the gradient of two bundles of data intervals. If this value is greater than the threshold value, it is considered the start of the peak, and when the mean gradient becomes a negative value, this point is put into the peak point candidate. It is expressed as follows:

$$\begin{aligned} G_n &= \frac{\text{SVM}_{n+1} - \text{SVM}_n}{T_{n+1} - T_n}, \\ \text{Average of } G_n &= \frac{G_n + G_{n+1}}{2}. \end{aligned} \quad (3)$$

The peak candidate includes waveform errors or noise errors. The following method is used to clear out the errors and find the genuine peaks. First, we find the peak candidates with a time interval of less than 0.3 seconds. Collected data are acceleration data for detecting the number of steps, so the movements must show regular intervals of high peak and low peak. Therefore, peak candidates in the low period are noise values from the wrong movement. Then, we store the candidate with high SVM values as the actual peak and drop the values considered as errors.

Detected peak values are affected by the patient's footsteps and the height of the swinging of arms, so the values include individual differences. However, every waveform of walking has high amplitude followed by low amplitude. Therefore, we use this feature to derive a threshold value with the mean amplitude over 1 second and collect the peaks over the threshold value. Figure 6 shows the result from the detection of step peaks.

*4.4. Results of Experiment.* The proposed algorithm is tested with the watch-type monitoring device with an embedded

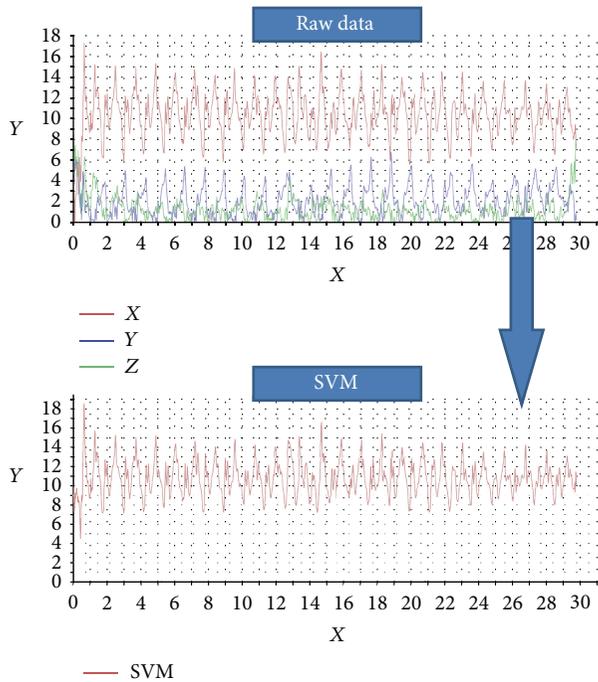


FIGURE 4: Preprocessing: convert raw data to SVM value.

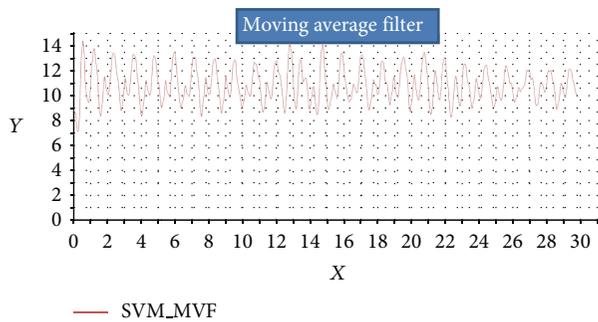


FIGURE 5: Preprocessing: moving average filter.

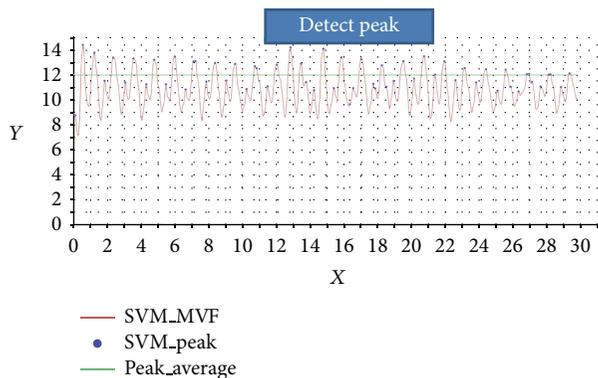


FIGURE 6: Result from the detection of step peaks.

accelerometer using an 80 Hz sample rate, attached to experimenters' wrists, and tested on fast steps, normal steps, and slow steps.

To measure the accuracy of the proposed algorithm, we compared the actual sum of steps and the detected sum of steps derived with the algorithm. The results of this method showed 94.7% accuracy in total, 93% in fast steps, 96.7% in normal steps, and 96% in slow steps.

As the pace gets faster, the gradient of SVM tends to grow larger and the phase interval narrows, resulting in higher error rates. However, in cases of normal and slow steps in which the amplitude is gradual, results have a higher rate of finding the peaks correctly, showing a closer value to the actual number of steps. Table 1 shows the analyzed data from the 8 people taking part in the experiment.

### 5. Patient Profile Management System

The purpose of this paper is to monitor daily health information to manage the dosage adjustment and health care of dementia patients. Measures of the amount of outdoor action and the resulting information on momentum can be health information. The patient profile management system profiles patient's daily information. Patient's daily information can be generated and the disappearance of the patient can be prevented through position information by integrating patient data received via a smart watch. In this paper, a function that analyzes patient's momentum and integrates received data is included to implement such a system.

The amount of exercise analysis calculates the number of steps measured by the acceleration sensor as momentum according to the rules. After the acceleration sensor data received from the smart watch is integrated with data about a patient's sex, age, weight, and height stored in the server, the integrated data generates momentum information.

*5.1. Amount of Exercise Analysis.* The step count obtained through the step detection algorithm can be used as data that measures momentum. The patient's data, which is basically stored in the server, includes age, height, weight, and personal information and this data is used as the standard for measuring a patient's stride and momentum.

The motion characteristics such as stationariness, walking and running, and information corresponding to moving distance and exercise time are needed in order to calculate the momentum. The moving distance can be measured through the GPS sensor, but it is difficult to measure the exact moving distance due to errors of the GPS sensor and the difference between indoors and outdoors. Therefore, the method that multiplies stride by the number of steps is used to calculate the patient's moving distance in this paper. The stride can be calculated by subtracting 100 from an individual's height, and momentum can be calculated as shown in below.

Amount of exercise

$$= \text{Amount of energy consumption (Kcal/ min *kg)}$$

$$* \text{Exercise per minute (min) * Weight (kg)}.$$

(4)

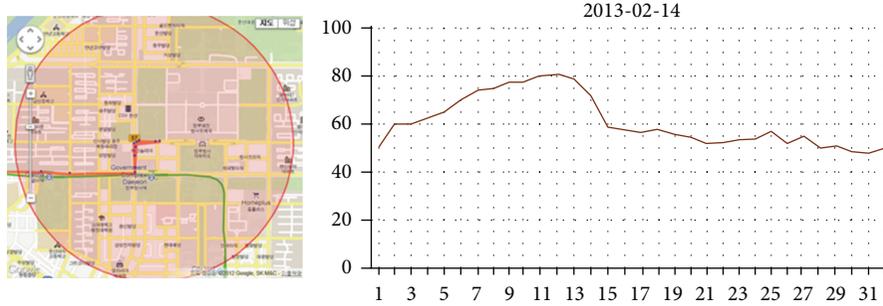
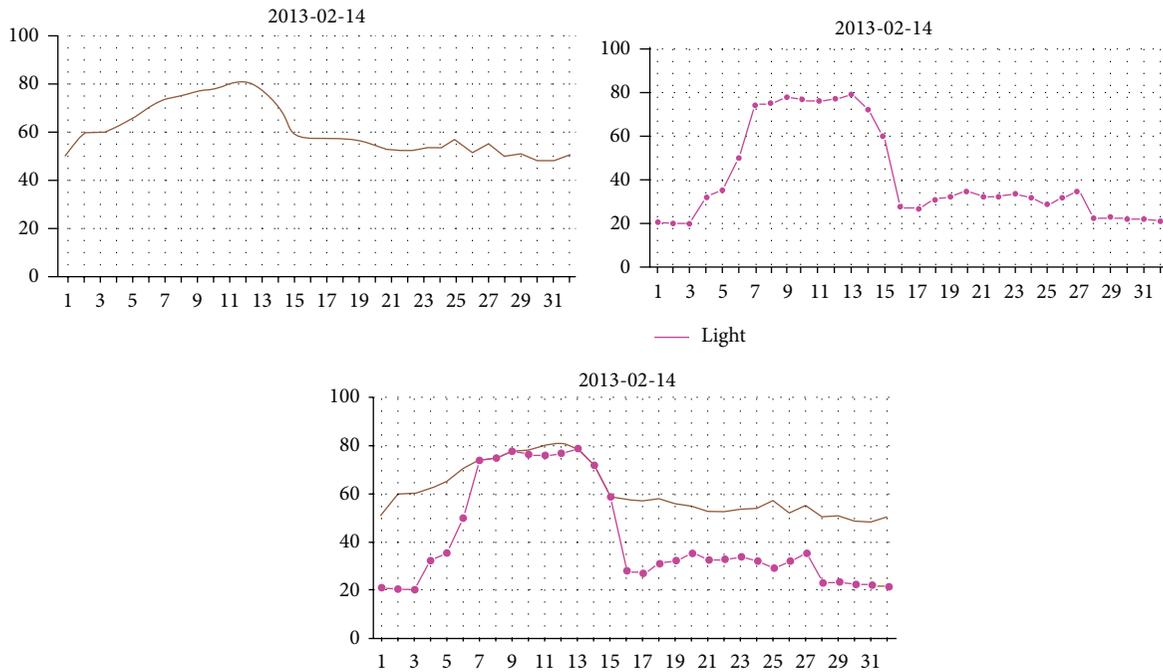


FIGURE 7: Function for monitoring: GPS, amount of activity.



Life information

◀ 1/22 ▶

| Tic      | 1  | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   | 11 | 12 | 13   | 14 | 15 | 16   | 17 | 18   | 19   | 20   |
|----------|----|------|------|------|------|------|------|------|------|------|----|----|------|----|----|------|----|------|------|------|
| Exercise | 51 | 60.2 | 60.3 | 65.4 | 70.9 | 74   | 75   | 77.8 | 78   | 80.4 | 81 | 79 | 72   | 59 | 58 | 57   | 58 | 56.2 | 55.1 | 54   |
| Light    | 21 | 20.2 | 20.3 | 32.3 | 35.4 | 50.0 | 73.3 | 77.5 | 78.3 | 76.2 | 76 | 77 | 77.2 | 56 | 28 | 27.2 | 31 | 32.2 | 32.6 | 29.2 |
| Outdoor  | N  | N    | N    | Y    | Y    | Y    | Y    | Y    | Y    | Y    | Y  | Y  | Y    | Y  | Y  | N    | N  | N    | N    | N    |

FIGURE 8: Patient profile system.

Energy consumption varies with motion characteristics and bottom surface. Table 2 shows energy consumption when walking on the basic asphalt.

The monitoring system developed in this paper can monitor a patient's momentum, current position, and the amount of light through a web page by using the GPS route information during outdoor activities, step detection, and momentum detection. Figure 7 shows the functions of monitoring server.

5.2. Create Patient Profiles. The patient's profile includes the patient's momentum, amount of light, and indoor and outdoor detection information by GPS. The patient's data is obtained in every cycle and the patient's momentum is calculated. The calculated result is integrated and then stored.

Figure 8 shows the screen applying the patient profile management system developed in this paper. The momentum obtained from the patient is divided into momentum, which is converted into a percentage and momentum converted into

TABLE 1: Experimental results.

|                | 58  | 71  | 72  | 83  | 99  | Lab no. |     |     | Total     | Accuracy |
|----------------|-----|-----|-----|-----|-----|---------|-----|-----|-----------|----------|
| Fast step      |     |     |     |     |     |         |     |     |           |          |
| U.C            | 117 | 111 | 111 | 109 | 116 | 118     | 117 | 105 | 904       | 93.03%   |
| R              | 138 | 120 | 119 | 117 | 121 | 123     | 120 | 109 | 967       |          |
| Slow step      |     |     |     |     |     |         |     |     |           |          |
| U.C            | 33  | 35  | 40  | 32  | 39  | 31      | 32  | 35  | 277       | 96.02%   |
| R              | 33  | 36  | 41  | 33  | 44  | 31      | 34  | 36  | 288       |          |
| Normal step    |     |     |     |     |     |         |     |     |           |          |
| U.C            | 71  | 77  | 71  | 72  | 68  | 66      | 66  | 68  | 559       | 96.77%   |
| R              | 77  | 75  | 66  | 72  | 75  | 61      | 73  | 78  | 577       |          |
| Total mean (%) |     |     |     |     |     |         |     |     | 1832/1740 | 94.71%   |

U.C.: user count—The number of steps counted by the user.

R: result of algorithm—The number of steps counted by the proposed algorithm.

TABLE 2: Amount of exercise on asphalt.

|        | 1 min | 2 min | 3 min | 10 min |
|--------|-------|-------|-------|--------|
| 50 Kg  | 4     | 8     | 12    | 120    |
| 60 Kg  | 3.8   | 9.6   | 14.4  | 144    |
| 70 Kg  | 5.6   | 11.2  | 16.8  | 168    |
| 80 Kg  | 6.4   | 12.8  | 19.2  | 192    |
| 90 Kg  | 7.2   | 14.4  | 21.6  | 216    |
| 100 Kg | 8.0   | 16    | 23    | 240    |

calories. After being integrated with light data, the profile can be developed of a patient's daily life. The patient's profile is updated daily. And it stores the daily information and moving route measured for a day. If the data is accumulated, the doctor can determine a more exact dosage and treatment method through the patient's daily life data.

## 6. Conclusion

In this paper, we developed an ubiquitous health management system for dementia patients following the concept of IoT. It is composed of a watch-type monitoring device and server that not only monitors patients' locations but also manages patients' health by determining patients' activity according to the data derived with the step detection algorithm, along with the ambient light sensor and accelerometer. According to the results of the experiments, normal steps have 96% accuracy in detection and on average showed 94% accuracy.

Typical medical services for dementia focused mainly on tracking the patients' location to prevent a patient from going missing or getting lost. The system developed in this paper provides and monitors the health information of the patients as well as location tracking. Further research based on this work could include a more comprehensive analysis of a patient's activities such as running or sitting and extensive application of the IoT paradigm.

## Conflict of Interests

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

## Acknowledgment

This research is supported by Seoul R&BD Program (SS110008).

## References

- [1] M. H. Tabert, X. Liu, R. L. Doty et al., "A 10-item smell identification scale related to risk for Alzheimer's disease," *Annals of Neurology*, vol. 58, no. 1, pp. 155–160, 2005.
- [2] M. Brahami, A. Baghdad Atmani, and A. Matta, "Dynamic knowledge mapping guided by data mining: application on healthcare," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 1–30, 2013.
- [3] Company Keruve, 2008, <http://www.keruve.com/>.
- [4] U-Safe Gangnam, 2009, <http://www.gangnam.go.kr/>.
- [5] KT I-Search, 2009, <http://www.kt.com/>.
- [6] G. E. Mead, W. Morley, P. Campbell, C. A. Greig, M. McMurdo, and D. A. Lawlor, "Exercise for depression," *The Cochrane Database System Reviews*, vol. 3, Article ID CD004366, 2009.
- [7] H. Y. Moon, S. H. Kim, Y. R. Yang et al., "Macrophage migration inhibitory factor mediates the antidepressant actions of voluntary exercise," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 32, pp. 13094–13099, 2012.
- [8] B. Kim, T. Kim, H.-G. Ko, D. Lee, S. J. Hyun, and I.-Y. Ko, "Personal genie: a distributed framework for spontaneous interaction support with smart objects in a place," in *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication (ICUIMC '13)*, Kota Kinabalu, Malaysia, January 2013.
- [9] R. S. H. Istepanian, S. Hu, N. Y. Philip, and A. Sungoor, "The potential of Internet of m-health things "m-IoT" for non-invasive glucose level sensing," in *Proceedings of the 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '11)*, pp. 5264–5266, Boston, Mass, USA, September 2011.

- [10] C. Doukas and I. Maglogiannis, "Bringing IoT and cloud computing towards pervasive healthcare," in *Proceedings of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS '12)*, pp. 922–926, Palermo, Italy, 2012.
- [11] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [12] J. Yang, "Toward physical activity diary: motion recognition using simple acceleration features with mobile phones," in *Proceedings of the 1st ACM International Workshop on Interactive Multimedia for Consumer Electronics (IMCE '09)*, pp. 1–10, New York, NY, USA, October 2009.
- [13] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Pervasive Computing*, vol. 3001 of *Lecture Notes in Computer Science*, pp. 1–17, Springer, Berlin, Germany, 2004.
- [14] J. Baek, G. Lee, W. Park, and B. J. Yun, "Accelerometer signal processing for user activity detection," in *Knowledge-Based Intelligent Information and Engineering Systems*, vol. 3215 of *Lecture Notes in Computer Science*, pp. 610–617, Springer, Berlin, Germany, 2004.
- [15] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI '05)*, vol. 20, pp. 1541–1546, Pittsburgh, Pa, USA, July 2005.
- [16] H. W. Yoo, J. W. Suh, E. J. Cha, and H. D. Bae, "Walking number detection algorithm using a 3-axial accelerometer sensor and activity monitoring," *Korea Contents Association Journal*, vol. 8, no. 8, pp. 253–260, 2008.
- [17] S. H. Shin, C. G. Park, J. W. Kim, H. S. Hong, and J. M. Lee, "Adaptive step length estimation algorithm using low-cost MEMS inertial sensors," in *Proceedings of the IEEE Sensors Applications Symposium (SAS '07)*, pp. 1–5, San Diego, Calif, USA, February 2007.
- [18] Y. H. Noh, S. Y. Ye, and D. U. Jeong, "System implementation and algorithm development for classification of the activity states using 3 axial accelerometer," *Journal of the Korean Institute of Electrical and Electronic Material Engineers*, vol. 24, no. 1, pp. 81–88, 2011.
- [19] Y. Luo, O. Hoerber, and Y. Chen, "Enhancing Wi-Fi fingerprinting for indoor positioning using human-centric collaborative feedback," *Human-centric Computing and Information Sciences*, vol. 3, article 2, pp. 1–23, 2013.
- [20] J. Ahn and R. Han, "An indoor augmented-reality evacuation system for the Smartphone using personalized Pedometry," *Human-centric Computing and Information Sciences*, vol. 2, article 18, pp. 1–23, 2012.