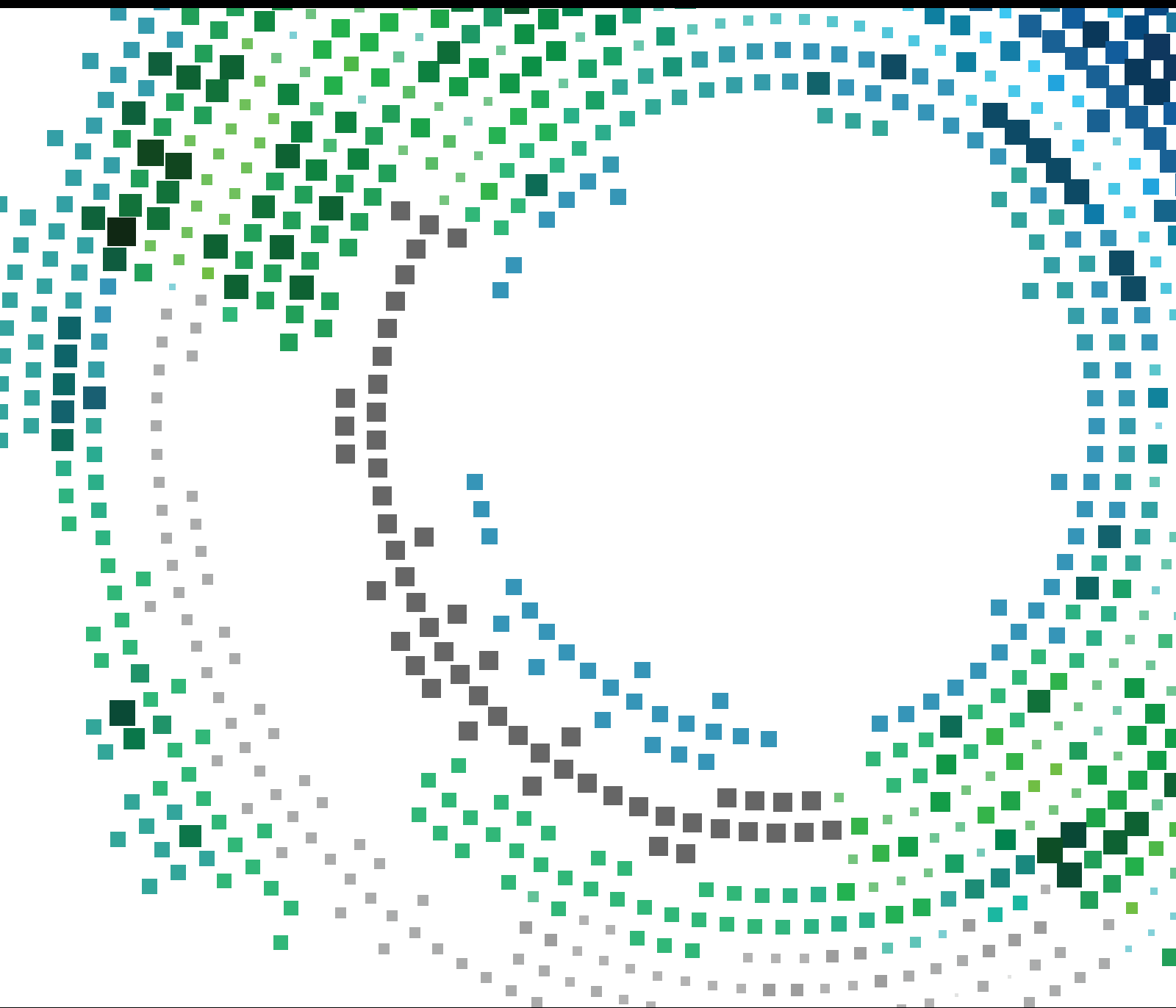
Mobile Information Systems

Location-Based Mobile Marketing Innovations

Lead Guest Editor: Jaegeol Yim
Guest Editors: Subramaniam Ganesan and Byeong H. Kang



Location-Based Mobile Marketing Innovations

Mobile Information Systems

Location-Based Mobile Marketing Innovations

Lead Guest Editor: Jaegeol Yim
Guest Editors: Subramaniam Ganesan and Byeong H. Kang

Editorial Board

Contents

Hindawi
Mobile Information Systems
Volume 2017, Article ID 1303919, 3 pages
https://doi.org/10.1155/2017/1303919

Editorial

Location-Based Mobile Marketing Innovations

Jaegeol Yim,[1] Subramaniam Ganesan,[2] and Byeong Ho Kang[3]

[1]*Department of Computer Engineering, Dongguk University, Gyeongju, Republic of Korea*
[2]*Electrical and Computer Engineering, Oakland University, Rochester, MI, USA*
[3]*School of Engineering and ICT, University of Tasmania, Hobart, TAS, Australia*

Correspondence should be addressed to Jaegeol Yim; yim@dongguk.ac.kr

Received 10 July 2017; Accepted 10 July 2017; Published 12 October 2017

The increasing complexity of the industry means that marketers must now be experts not only in marketing but also in people, data, delivery platforms, and mobile location-based marketing. The objective of location-based marketing via mobile devices is to encourage those activities as well as drive foot traffic, share discounts, and build customer loyalty. Mobile devices have been used to gather information about nearby businesses including reviews, directions, calling the business, and using the businesses' mobile app. With location-based mobile marketing, the business is easy to find and have skillfully combined location-based marketing with an overall targeted marketing approach that includes social media, push notifications, email newsletters, and also offline marketing.

The objective of this special issue is to bring together research contributions of unpublished research on the recent development and innovations about location-based mobile marketing (LBMM). The fundamental infrastructure of LBMM is obviously location-based service (LBS) systems. The most distinguished parts of LBS systems include positioning components and map provisioning components that are running on the general computing systems. On top of the LBS system, LBMM requires providing more intelligent marketing oriented services to customers. More and more machine learning methods, especially convolutional neural networks, have been used to analyze user behavior and classify usage patterns in development of intelligent services. With increasing model training parameters and data scales, the traditional single machine training method cannot meet the requirements of time complexity in practical application scenarios. Therefore, the data parallel model or parallel method has been employed in order to speed up the training process. The current training framework often uses simple data parallel or model parallel methods that do not fully utilize heterogeneous computing resources. The paper entitled "A Parallel Strategy for Convolutional Neural Network Based on Heterogeneous Cluster for Mobile Information System" by J. Zhang et al. proposes a delayed synchronization convolutional neural network using the heterogeneous system. This is a parallel strategy based on both synchronous parallel and asynchronous parallel approaches, and the model training process can reduce the dependence on the heterogeneous architecture in the premise of ensuring the model convergence. Therefore, the convolution neural network framework is better suited for different heterogeneous system environments.

Intelligent service is to guess desired services that the customer might want and recommend them to the customer. The SlopeOne model has been widely used in guessing desired services. The paper entitled "Collaborative QoS Prediction for Mobile Service with Data Filtering and SlopeOne Model" by Y. Yin et al. proposes a data filtering-extended SlopeOne model (based on collaborative filtering). The idea behind this model is based on the characteristics of a mobile service and the relation with location. With the model, the quality of service (QoS) values can be predicted.

As an important source of mobile location-based data, taxi mobility information can be referred to when making marketing decisions. Studying the behavioral patterns of taxis in a city during the holidays using the global positioning system (GPS) can yield remarkable insights into people's holiday travel patterns, as well as the odd-even day vehicle prohibition system. Using GPS data, the paper entitled "Exploring Intracity Taxi Mobility During Holidays for Location-Based

Marketing" by W. Wang et al. studied the behavioral patterns of taxis during specific holidays in terms of pick-up and drop-off locations, travel distance, mobile step length, travel direction, and radius of gyration.

LBMM gathers and uses information about users' locations. Therefore, ensuring information security and privacy is one of the most important concerns of users. The paper entitled "A Hybrid Location Privacy Solution for Mobile LBS" by R. Gupta and U. P. Rao proposes a hybrid solution, HYB, to achieve location privacy for the mobile users who use location services frequently. The proposed HYB scheme is based on the collaborative preprocessing of location data and utilizes the benefits of homomorphic encryption technique. Location privacy is achieved at two levels, namely, at proximity level and at distant level. The proposed HYB solution preserves user's location privacy effectively under specific, pull-based, sporadic query scenario.

LBMM systems and as an example the sensor nodes often face various attacks where the attackers try to manipulate the estimated location or try to provide false beacons. The paper entitled "A Secure Localization Approach Using Mutual Authentication and Insider Node Validation in Wireless Sensor Networks" by G. Kumar et al. has proposed a methodology that will address this problem of security aspects in localization of the sensor nodes. Moreover, they have considered the network environment with random node deployment and mobility as these two conditions are less addressed in previous research works. Further, they proposed an algorithm that requires low overhead due to the usage of less control messages in a limited transmission range. In addition, they have also proposed an algorithm to detect the malicious anchor nodes inside the network. The paper entitled "Location Privacy Protection Based on Improved K-Value Method in Augmented Reality on Mobile Devices" by C. Yin et al. proposes a privacy protection method combining the k-anonymity with pseudonym methods and shows that the method can effectively anonymize all service requests. The paper titled "An Enhancement of Optimized Detection Rule of Security Monitoring and Control for Detection of Cyber Threat in Location-Based Mobile" by W. Park and B. H. Kang analyzes SNORT detection rules and proposes a guideline of SNORT rule optimizations to improve the efficiency and accuracy of intrusion detection operations. This enables the intrusion detection system to be more secure in location-based mobile services.

In the mobile service environment, it is required that mobile terminals should efficiently use wireless network resources. In addition, because video streaming becomes a major service among the data services of mobile terminals in heterogeneous networks, the necessity of the efficient network access control for heterogeneous wireless networks is raised as an important topic. The paper titled "Network Access Control for Location-Based Mobile Services in Heterogeneous Wireless Networks" by D.-Y. Kim et al. proposes a novel network access control in heterogeneous wireless networks. The proposed method estimates the network status with Naïve Bayesian Classifier and performs network access control according to the estimated network status. Thus, it improves data transmission efficiency to satisfy the quality of services.

Mobility management is an essential challenge for supporting reliable multimedia data streaming over wireless and mobile networks in the Internet of Things (IoT) for LBMM applications. The paper titled "Design of Optimized Multimedia Data Streaming Management Using OMDSM over Mobile Networks" by B. Park et al. introduces a new enhanced data streaming route optimization scheme that uses an optimized Transmission Control Protocol (TCP) realignment algorithm in order to prevent the packet disordering problem whenever the nodes in the IoT environment are communicating with each other. With the proposed scheme, data packets sequence realignment can be prevented, the packet traffic speed can be controlled, and the TCP performance can be improved.

In large-scale location-based services, an ideal situation is that self-adapting routing strategies use future traffic data as input to generate a topology which could adapt well to the changing traffic. The paper entitled "A Traffic Prediction Model for Self-Adapting Routing Overlay Network in Publish/Subscribe System" by M. Chi et al. proposes a traffic prediction model for the broker in publish/subscribe system, which can predict the traffic of the link in future by using neural networks.

As LBMM systems evolve and become more and more intelligent, they require more and more complicated operations and computing resources. As a result, executing them on the mobile computing systems takes too much time or becomes impossible. To overcome this problem, a computation offloading technique can be used to execute certain tasks of LBMM in cloud and fog environments. The paper entitled "Offloading Method for Efficient Use of Local Computational Resources in Mobile Location-Based Services Using Clouds" by Y. Son and Y. Lee introduces a computation offloading technique that utilizes fog computing to improve the performance of virtual machines running on mobile devices. LBMM systems make use of virtual machine technology in order to run on a variety of platforms.

LBMM systems, with most components of them running on mobile terminals with limited computing resources, should be optimized in terms of resource consumption—especially battery power. The paper entitled "Automatic Optimizer Generation Method Based on Location and Context Information to Improve Mobile Services" by Y. Son et al. introduces a technique to automatically generate a customized service optimizer for each application, service type, and platform using location and situation information. By using the proposed technique, energy and computing resources can be more efficiently employed for each service. Thus, users should receive more effective LBSs on mobile devices, such as smartphones.

If we can detect the difference between two business process models, we can quickly build a new model by adjusting the existing one. The paper entitled "Detecting Difference between Process Models Based on the Refined Process Structure Tree" by J. Fan et al. presents a new approach that detects the difference of the entire composition. Firstly, it parses the process models to their corresponding

refined process structure trees (PSTs). Then it converts the PSTs into their corresponding task based process structure trees (TPSTs). As a consequence, the problem of detecting differences between two process models is transformed to detect difference between two TPSTs. Finally, it obtains the differences between two TPSTs based on the divide-and-conquer strategy, where the differences are described by an edit script and it makes the cost of the edit script close to minimum.

New information technologies including sensors, indoor positioning, augmented reality, vision, and interactive systems should be utilized by retailers in order to improve operational efficiency and customer experience. The paper entitled "Use of the Smart Store for Persuasive Marketing and Immersive Customer Experiences: A Case Study of Korean Apparel Enterprise" by H. Hwangbo et al. employs the term "smart store" to indicate retail stores equipped with these new technologies and modern marketing concepts. They summarize discussions related to smart stores and their possible applications in a real business environment. Furthermore, they present a case study of a business that applies the smart store concept to its fashion retail shops in Korea.

Acknowledgments

Yim's work was supported by Small and Medium Business Association (C0443742), by Dongguk University Research Fund, and by the Ministry of Knowledge Economy (10037393).

Jaegeol Yim
Subramaniam Ganesan
Byeong Ho Kang

Hindawi
Mobile Information Systems
Volume 2017, Article ID 8501976, 13 pages
https://doi.org/10.1155/2017/8501976

Research Article

An Enhancement of Optimized Detection Rule of Security Monitoring and Control for Detection of Cyberthreat in Location-Based Mobile System

Wonhyung Park[1,2] and Byeong Ho Kang[1,2]

[1]*Department of Industrial Security, Far East University, Gamgok-myeon, Eumseong-gun, Chungcheongbuk-do 369-700, Republic of Korea*
[2]*School of Engineering and ICT, University of Tasmania, Private Bag 87, Hobart, TAS 7001, Australia*

Correspondence should be addressed to Byeong Ho Kang; bhkang@utas.edu.au

Received 9 December 2016; Revised 27 June 2017; Accepted 6 July 2017; Published 19 September 2017

Academic Editor: Floriano Scioscia

A lot of mobile applications which provided location information by using a location-based service are being developed recently. For instance, a smart phone would find my location and destination by running a program using a GPS chip in a device. However, the information leakage and the crime that misused the leaked information caused by the cyberattack of mobile information system occurred. So the interest and importance of information security are increasing. Also the number of users who has used mobile devices in Korea is increasing, and the security of mobile devices is becoming more important. Snort detection system has been used to detect and handle cyberattacks but the policy of Snort detection system is applied differently for each of the different kinds of equipment. It is expected that the security of mobile information system would be improved and information leakage would be blocked by selecting options through optimization of Snort detection policy to protect users who are using location-based service in mobile information system environment in this paper.

1. Introduction

The importance of location-based services (LBS), which is a wired and wireless Internet service utilizing current and past location information of users with terminal which can track location, is emphasized due to the development of mobile communication technology and the rapid spread of mobile terminals [1].

The location-based service is a service that identifies the user's location using Location Detection Technology and adds related applications. I think it can be used for various purposes, creating added value using application and location information of wired and wireless Internet.

Also, due to the recent development of cyberattack technology, information leakage as hacking and personal information exposure has become a problem. There is a high concern about exposure of personal information to the current location due to the nature of location information services. In the member information exposed through the online site, personal information such as a name, a resident registration number (ID), an address, and a resident registration number may be used for other purposes through theft. Further, the location information of the customer and the identification of the movement trajectory through the location information may already act as direct privacy violation factors. For this reason, concerns about privacy breaches caused by leakage of location-based services are more serious in Korea [2–4].

The key to security monitoring is rapid detection of cyberattacks. Among the various security monitoring systems, a network-based intrusion detection system (IDS) is the only system that can detect application attacks such as

Table 1: Location-based service utilization [5].

Sort	Field of application	Benefit
(1)	Tracking the location of a young or demented elderly	Missing child prevention, accident prevention
(2)	Tracking your pet location	Lost, accident prevention
(3)	Vehicle navigation	Identifying the route of the vehicle
(4)	Location of field rep	Effective management of field rep
(5)	Providing information about the current location	Nearby information services such as theaters, gas stations, restaurants
(6)	Police, security, military vehicle management	Crime prevention
(7)	Providing location information of courier and cargo	Reducing oil, transportation, and communication costs

web hacking most efficiently by installing them between the control network entrances. The function of the intrusion detection system (IDS) is to use a pattern matching method that detects an attack and generates an alarm when a header or Payload information communicating through the network is detected as an attack.

However, if an attacker encrypts communication signals due to attack packets or malicious code infections, the intrusion detection system (IDS) only checks the encrypted packets. Even if the attack packet is an actual attack packet, it cannot be detected and waypoint also cannot be detected. In order to detect such an attack, it is necessary to develop a behavior-based detection system that can detect and alert an attack using an unknown attack technique instead of a pattern matching methods [6–8].

Currently, security monitoring technology analyzes cyberattack techniques and malicious codes, extracts patterns such as certain strings, and then uses this pattern to develop detection patterns (signature) and apply them to intrusion detection systems. After that, if the cyberattack information matches the detection pattern, it is detected as an accident. If the attack technique is changed, the detection pattern should be corrected in a timely manner so as to maintain the optimized state. However, it is not easy to detect new attacks or malicious codes.

In addition, recent cyberattack techniques such as hacking and distribution of malicious code are developing rapidly and utilizing advanced and intelligent techniques such as double encryption technique to prevent detection by security monitoring or vaccine. So it is not enough to completely detect and block new cyberattacks.

Therefore, in order to efficiently detect and respond to cyberattacks in systems that utilize location-based services in mobile information systems, it is necessary to optimize security monitoring detection techniques to share information among security monitoring centers or to standardize detection patterns according to heterogeneous equipment.

2. Related Work

2.1. Location-Based Service. LBS is an acronym for location-based service. It is generally defined as an application system and service that accurately grasps the location of a person or object based on the mobile communication network and utilizes it. Accordingly, the LBS is a system that grasps the location information of an individual or a vehicle through a mobile communication base station and a GPS (Global Positioning System) and provides various advanced services based on the information [1].

LBS provides various application services based on location information. These include emergency assistance, location information services, traffic congestion and navigation information, and location-based billing. Other applications include Intelligent Transport Systems (ITS), assistive devices for people with disabilities, L-Commerce based on location information, and cell ID-based friends using cell phones (see Table 1).

The current location information acquisition technology of the wireless communication network enables collecting more precise location information by combining the GPS and other location positioning technology and wireless communication network, and it is possible to provide more various application services. As the location information is connected with the mobile communication network, it is possible to provide a general service in the future, and the application service structure provided in the network is changing from a wired/wireless communication network structure with an independent vertical structure to a horizontal structure for wired/wireless integration. Also, all network entities will evolve into an open converged network that provides services based on an equalized All-IP network. Through the development of position location system such as A-GPS (Assistance-GPS) and the paradigm change of ubiquitous and pervasive computing environment, MT (Mobile Terminal) will become a subject of information provision independently and will develop its form to deliver its location information to LBS SP (Service Provider). With these developments, it is necessary to provide the components of location-based services with safety and reliability beyond the conventional wired and wireless network level.

2.2. Intrusion Detection System. Intrusion detection system (IDS) was introduced in 1980 by James Enderson of the United States in a paper called "Computer Security Threat Monitoring and Surveillance." In 1986, Dorothy Denning published an article entitled "An Intrusion Detection Model" and was influenced by IDS.

Intrusion detection systems can reduce the misuse detection and improve the performance of the system by designing efficient and complete detection rules for cyberthreats. Rules

should be as simple and flexible as possible and handle large amounts of network traffic without packet loss. This requires testing procedures to assess the appropriateness before applying the developed rules and periodic optimization to speed up the rules.

For exact detection rules, you must test them before applying them in the intrusion detection system (IDS). Inaccurate rules cause too many false positives and false negatives. A large number of false detection events may cause unnecessary analysis time, prevent detection of normal attack events, or cause the network sensor of the IDS to go down. In order to reduce false detection events, test procedures are required before the system is applied. When testing, efficiency, usability, accuracy, and uniqueness should be considered.

In addition, false positives should be reduced. False positive events occur when you configure detection rules extensively or when you activate unnecessary rules. In order to reduce this, we need to rigorously apply detection rules through precise analysis of the exploit. In addition, it disables the detection rules of the simple information providing format such as "ICMP UNREACH" to reduce the load of the cyberthreat attack event. Inaccurate rules flood false positives and generate false negatives. A large number of false detection events may cause unnecessary analysis time and may prevent detection of cyberthreat attack events.

Until now, the term "security monitoring" has not been defined as a legal rule. In recent years, it has been a step in the process of conceptualization in the academic sense. The term "security control" is used in English as "Security Monitoring" or "Security Monitoring & Control." The dictionary meaning of "Monitoring" is to protect against various errors that may occur during computer program execution. And the Korean dictionary of the Korean language states that "control" means "to control and control by necessity at a country or an airport" [9] (see Figure 1).

2.3. Intrusion Detection System. The Snort intrusion detection system is one of the most widely used systems among intrusion detection systems (IDS) and is an open source network-based intrusion detection system (open source NIDS) [12–14].

The rule is divided into Header and Option. As shown in (Figure 2), detailed rules can be distinguished as conditions to be detected in the detection operation, protocol type, source address, source port, traffic transmission direction, destination IP address, and destination port. The elements used in these detailed rules are summarized as shown in Figure 2.

The Rule Header of Snort is an integral part of the detection rule that includes five elements: Rule Action, Protocol, Source, Destination IP, Source, Destination Port, and Traffic Direction. Rule Action specifies what the rule should do if the packet matches the rule. Snort has rule actions such as "pass, log, alert," but in most cases it uses the alert Rule Action [15–18] (see Table 2).

Snort's rule options are divided into General, Payload Detection, and Nonpayload Detection rule options as shown in Table 3.

3. Optimization of Selected Snort-Based Detection Rule

3.1. Header Detection Rule Optimization. In Rule Action, "alert" generates a warning, "log" leaves a log, "pass" ignores the packet, "activate" sends a warning and activates the specified dynamic rule, and "drop" throws away the packet and leaves a log. Also "reject" leaves the connection and log, and "sdrop" discards the packet and leaves no log. Of these, 6 items including "log," "pass," "activate," "dynamic," "reject," and "sdrop" are excluded. For this reason, "log" and "pass" are options for packet logging or packet ignoring. "Activate" and "dynamic" are used mainly for additional logging after detection of attack. They are not suitable for the purpose of notifying the occurrence of attack. "Reject" and "sdrop" are excluded because they are additional actions after interception.

In the protocol, "tcp," "udp," "icmp," and "ip" support the TCP, UDP, ICMP, and IP protocols, respectively. In the protocol, "tcp" supports the TCP protocol, "udp" supports the UDP protocol, "icmp" supports the ICMP protocol, and "ip" supports the IP protocol (see Table 4).

In IP, "any" represents All-IP address targets, "numeric IP" represents a specific IP address target, "numeric IP list" supports up to 10, including CIDR among multiple IP addresses, "CIDR" represents the length object of a specific network address, and "negation(!)" represents All-IP address destinations except the specified IP address. In port, "any" represents all port number targets, "static port" represents fixed port number targets, "ranges(:)" represents port range targets, and "negation(!)" represents all port destinations except for specified ports. In Direction, "-> option" indicates the direction of the destination host from the source host, and "<> option" indicates the direction of both the source host and the destination host. The <- option lowers the detection efficiency by generating a lot of intrusion detection sensor load. Also, "<-" is to remove the mandatory option because it is necessary to use the -> option by changing the source IP and destination IP (see Table 5).

3.2. General Rule Optimization. In General, "msg" is used as an option to indicate a message to be recorded when detecting security control events. "Reference" is a reference to additional information, "gid" is the ID of the alert generation module, sid is used to identify the Snort detection rule, "<100" is the number reserved for future use, "100–1,000,000" indicates the number assigned by Snort, and ">1,000,000" represents a user-defined rule assignment number. "Rev" keyword indicates information about the revision of the sid, "classtype" identifies information that can classify the attack, and "priority" indicates the importance of the rule. In General, all options excluding "msg" are excluded. The "reference" case is excluded as an additional option for reference of detection rule information. "Gid" and "sid" are excluded

Table 2: Definition of Snort Header detection rules [4].

Snort instruction format	Definition
Header	
Rule Action	
alert	Generate Alert
log	Leave log
pass	Ignore pat
activate	Send alerts and activate dynamic rules
dynamic	It is activated by the activate rule and the Log option
drop	Drop a packet and leave a log
reject	Connection terminated and logged
drop	Discard packets and leave no logs
Protocol	
tcp	TCP protocol support
udp	UDP protocol support
icmp	ICMP protocol support
ip	IP protocol support
IP	
any	All IP address
numeric IP	Specific IP addresses
numeric IP list	Multiple IP addresses
CIDR	Specific network class destination (i) Class A Network (8 bits) (ii) Class B Network (16 bits) (iii) Class C Network (24 bits)
negation(!)	All IP addresses except the specified IP address
Port	
any	All port numbers
static port	Fixed Port Number
ranges(:)	Port range destination
negation(!)	All ports except the specified port
Direction	
->	From the origin host to the destination host
<-	Change the source and destination information and specify to "->"
bidirectional(<>)	Bidirectional detection support

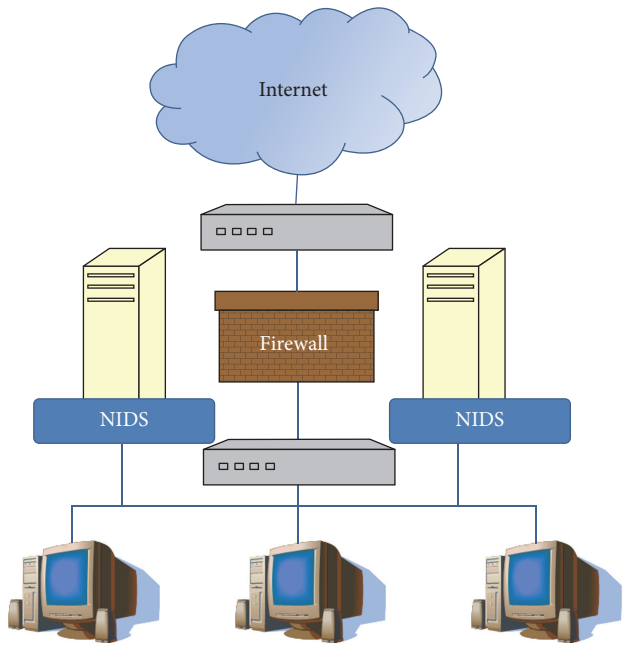


Figure 1: NIDS network [9].

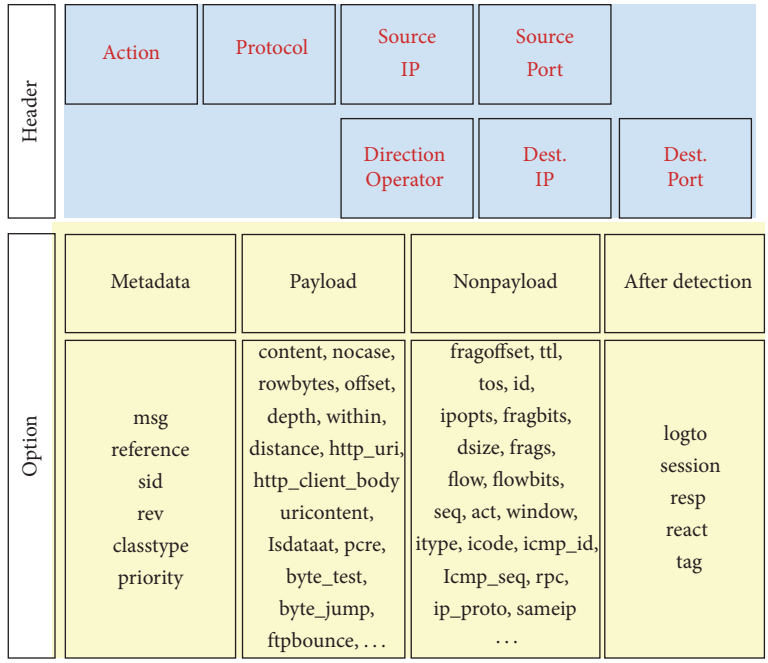


Figure 2: Snort basic rule set [10, 11].

from Snort configuration module as indicating module ID and detection rule ID that generated warning. Also, "rev" is excluded as an option for version control of detection rules, and "classtype" and "priority" are excluded due to lack of usability as an option for sorting and prioritizing detection rules (see Table 6).

3.3. Payload Detection and Nonpayload Detection Rule Optimization. In Payload Detection (Content, Content Modifier), "content" indicates the specific content to be found in the Payload of the packet, and "nocase" means not case sensitive. "Rawbytes" ignores the decoding process and indicates raw packet data inspection, offset indicates the pattern search start position, depth indicates the pattern search range, distance indicates a new pattern search start position after the previous pattern matching, and within indicates the pattern search range. The http_client_body searches in the body part of the HTTP request. The http_cookie searches in the cookie part of the HTTP header. The http_header searches in the HTTP header part. The http_method searches in the HTTP method part. The http_uri part searches the HTTP URI part in the fast_pattern Eye. This is the command to designate the pattern to search first. However, HTTP related commands can be specified with the content option and can be excluded. Fast_pattern excludes string matching as a priority (see Table 7).

In Payload Detection, "uricontent" searches patterns from URI information of HTTP, "urilen" checks HTTP URI length, and "isdataat" searches whether Payload has a certain number of bytes. "Pcre" searches for a regular expression, byte_test compares it to a specific value after a certain byte operation, and "byte_jump" jumps to a result value after a certain byte operation. "Ftpbounce" detects an FTP bounce attack, "asn1" detects a malicious encoding, and "cvs" detects an invalid entry string in CVS. Also, "dce_iface," "dce_opnum," and "dce_stup_data" detect the DCE/RPC request traffic pattern. Of these, "urilen" is excluded because it can be specified using mandatory options, and "ftpbounce," "asn1," "cvs," "dce-iface," "dce-opnum," and "dce_stup_data" commands should be excluded because these are the options for detecting specific attacks on specific services (see Table 8).

Among the Nonpayload Detection options, the commands related to IP such as fragoffset, fragbits, tos, id, ipopts, and TCP related commands seq, ack, and windows are excluded because they are not useful in creating detection rules (see Table 9).

In Nonpayload Detection, "dsize" checks packet payload size to detect packets of abnormal size, and "flow" defines packet direction in relation to client-server communication stream. "Flowbits" is an option to support session-based detection, and "Rpc" acts to identify the rpc service but it is excluded because it can be specified using mandatory options. The "sameip" checks whether the source and destination IPs are the same, and the "stream size" checks the size of the session according to the TCP sequence number, but it is excluded because it can be specified through the "dsize" option. In Rule Thresholds, "Limit" indicates the first occurrence of a warning when a number of identical events occur within a certain time, and "Threshold" indicates a warning when the number of the same events occurring

Table 3: Definition of Snort option detection rules [4].

Snort instruction format	Definition
Option	
General	
msg	Message to record when Alert or logging
reference	References to additional information
gid	Alert generation module id
sid	Use to distinguish snort detection rules
rev	Display information about revision of rule with sid
classtype	Information that can classify an attack
priority	Show the importance (priority) of detection rules
Payload Detection	
content	Specific content looking for in the payload of a packet
content modifier	
nocase	Not classifying capital and small letter
rawbytes	Ignore the decoding process and check the raw packet data
offset	Specify whether to start pattern search after the first few bytes of the packet
depth	Specify how to compare pattern search from offset to how many bytes
distance	Specify whether to start pattern after how many bytes from previous pattern matching.
within	Specify how to compare pattern searches from distance to how many bytes
http_client_body	Search in body part of HTTP request
http_cookie	Search in the cookie portion of the HTTP header
http_header	Search in the HTTP header section
http_method	Search in the HTTP methods section
http_uri	Search in the HTTP URI section
fast_pattern	Specify the pattern to search first
uricontent	Retrieve patterns from URI information in HTTP
urilen	Check HTTP URI length
isdataat	Checks if the payload has a certain number of bytes
pcre	Search by regular expression
byte_test	Compare with specific value after specific byte operation
byte_jump	Jump as much as the operation result value after a certain byte operation
ftpbounce	FTP bounce attack detection
asn1	Detect malicious encoding
cvs	Detect invalid Entry string in CVS
dce_iface	
dce_opnum	Detect traffic pattern requesting DCE/RPC
dce_stup_data	
Non-Payload Detection	
IP	
fragoffset	IP fragment offset field check
fragbits	IP fragment offset field check
tos	IP Service type field check
id	IP identification field check
ttl	IP Time To Live field check
ip_proto	IP protocol inspection
ipopts	IP Options field check
TCP	
seq	TCP sequence number check
ack	TCP acknowledge number check
flags	TCP flag bit field check
window	TCP window size check

Table 3: Continued.

Snort instruction format	Definition
ICMP	
itype	ICMP type check
icode	ICMP code check
icmp_id	ICMP identification check
icmp_seq	ICMP sequence number check
dsize	Detect the payload size of packets to detect abnormal size packets
flow	Defines the direction of the packet in relation to the client-server communication stream
flowbits	Options to support session-based detection
rpc	rpc service identification
sameip	Check if origin and destination IP are the same
stream_size	Check the size of the session according to the TCP sequence number
Thresholding	
limit	Only the first warning occurs when multiple identical events occur within a certain time
threshold	Alert when the number of the same events that occur within a certain time is exceeded

Table 4: Optimization of Header Rules: Rule Action, Protocol.

Command format		Selection of detection rule standardization	
Rule Action	alert	Generate a warning	
	drop	Drop the packet and leave a log	
Protocol	tcp	TCP protocol support	
	udp	UDP protocol support	
	icmp	ICMP protocol support	
	ip	IP protocol support	
Command format		Excluded detection rules standardized/excluded reasons	
Rule Action	log	Logged	It is an option for packet logging or packet override, which is mainly used for logging after attack detection, but it is for the purpose of notifying the occurrence of an attack
	pass	Ignore packets	
	activate	Send an alert and activate the specified dynamic rule	This option is used for additional logging after detection of an attack, but it is consistent with the purpose of notifying the occurrence of the attack
	dynamic	It is activated by the activate rule and acts like the log option	
	reject	Connection terminated and logged	Added after Intrusion rrevention and exclude as action
	sdrop	Discards packets and leaves no logs	

Table 5: Optimization of Header Rules: IP, Port, Direction.

Command format		Selection of detection rule standardization
IP	any	All IP address
	numeric IP	Specific IP addresses
	numeric IP list	Multiple IP address up to 10 including CIDR
	CIDR	The length of a specific network address.
Port	any	all port numbers
	static port	Fixed Port Number
	ranges(;)	Port range destination
Direction	->	Direction from the origin host to the destination host
	<>	Origin host and destination host bidirectional
Command format		Excluded detection rules standardized/excluded reasons
Direction	<-	Source Host and Destination Host Reverse / It is excluded because it can be made by changing source IP and destination IP and generate load

Table 6: Optimization of General Rules.

Command format		Selection of detection rule standardization	
General	msg	Message to record when detecting	
Command format		Excluded detection rules standardized/excluded reasons	
General	reference	References to additional information	Excluded as an additional option for reference of detection rule information
	gid	Alert generation module id	
	sid	Use to distinguish Snort detection rules <100 reserved number for future use 100–1,000,000 number assigned by Snort >1,000,000 custom rule assignment numbers	Except for the module ID of the configuration module and the ID of the detection rule (Snort-specific function)
	rev	Information on revision of rules with sid	Excluded as an option for versioning of detection rules
	classtype	Information that can classify an attack	Excluded as an option for risk display and classification of detection
	priority	Significance of detection rules (top/middle/bottom)	Exclude as an option for indicating the importance of detection rules

Table 7: Optimization of Payload Detection (Content, Content Modifier) Rules.

Command format		Selection of detection rule standardization	
Payload Detection	content	Specific content to look for in the payload of a packet	
	nocase	Case insensitive	
	rawbytes	Ignore the decoding process and check raw packet data	
	offset	Pattern search start position (after the first few bytes of the packet)	
	depth	Pattern search range (compare pattern search from offset to several bytes)	
	distance	New pattern search start position after a previous pattern match (after a few bytes)	
	within	Pattern search range (compare pattern search from distance to several bytes)	
Command format		Excluded detection rules standardized/excluded reasons	
Payload Detection	http_client_body	Search in body part of HTTP request	Except for the content option
	http_cookie	Search in the cookie portion of the HTTP header	
	http_header	Search in the HTTP header section	
	http_method	Search in the HTTP methods section	
	http_uri	Search in the HTTP URI section	
	fast_pattern	Specify the pattern to search first	Excluded as string matching from specified priority

Table 8: Optimization of Payload Detection Rules.

Command format	Selection of detection rule standardization	
Payload Detection	isdataat	Check if the payload has a certain number of bytes
	pcre	Search by regular expression
	byte_test	Compare with specific value after specific byte operation
	uricontent	Search patterns from URI information in HTTP
Command format	Excluded detection rules standardized/excluded Reasons	
Payload Detection	urilen	Check HTTP URI length / Excluded as assignable opting using mandatory option
	ftpbounce	FTP bounce attack detection / Excluded as assignable opting using mandatory option
	asn1	Detect malicious encoding / Excluded as assignable opting using mandatory option
	cvs	Detect invalid Entry String in CVS / Excluded as assignable opting using mandatory option
	dce_iface	DCE/RPC request traffic pattern detection / Excluded as assignable opting using mandatory option
	dce_opnum	DCE/RPC request traffic pattern detection / Excluded as assignable opting using mandatory option
	dce_stup_data	DCE/RPC request traffic pattern detection / Excluded as assignable opting using mandatory option

Table 9: Optimization of Nonpayload Detection Rules 1.

Command format	Selection of detection rule standardization	
Nonpayload Detection (IP)	ttl	Inspect IP Time-To-Live field
	ip_proto	Inspect IP protocol field
Nonpayload Detection (TCP)	flags	Inspect TCP flag bit field
Nonpayload Detection (ICMP)	itype	Inspect ICMP type
	icode	Inspect ICMP code
	icmp_id	Inspect ICMP identification field
	icmp_seq	Inspect ICMP sequence number
Command format	Excluded detection rules standardized/excluded reasons	
Nonpayload Detection (IP)	fragoffset	Inspect IP fragment Offset field / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	fragbits	Check whether IP fragmentation and reserved bits are set / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	tos	Inspect IP Service type field / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	id	Inspect IP identification field / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	ipopts	Inspect IP Options field / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
Nonpayload Detection (TCP)	seq	Inspect TCP Sequence number / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	ack	Inspect TCP acknowledge number / It is excluded through consultation with related companies, Because it is not useful in creating detection rule
	window	Inspect TCP window size / It is excluded through consultation with related companies, Because it is not useful in creating detection rule

Table 10: Optimization of Nonpayload Detection Rules 2.

Command format	Standardization of detection rules Candidates for selection		
Nonpayload Detection	dsize	Packet detection of abnormal size by checking the packet's payload size	
	flow	Defines the direction of the packet in relation to the client-server communication stream	
	flowbits	Options to support session-based detection	
Rule Thresholds	Limit	Alert for the first time when multiple identical events occur within a certain time	
	Threshold	Alert when the number of the same events that occur within a certain time is exceeded	
Command format	Excluded detection rules standardized/excluded Reasons		
Nonpayload Detection	rpc	Identify the rpc service	It identifies the rpc service, but it can be specified using mandatory options. It can be specified through the dsize option.
	sameip	Check if origin and destination IP are the same	
	stream size	Check the size of the session according to the TCP sequence number	

within a certain time exceeds the corresponding number. Threshold option was used before Snort 2.8.5 version; Snort 2.8.5.1 or later uses Detection Filter or Event Filters option (see Table 10).

4. Comparison Analysis of Existing Snort Detection Options

Optimizing the existing Snort detection grammar will allow the user to understand and analyze the wrong type of policy created without considering the performance and false positives of the detection sensor in the event of a vulnerability attack. Also this can elaborate detection rules. In order to normalize the detection rules; first, if short strings are applied, frequent detection of the intrusion detection system sensor occurs, thereby degrading the performance of the intrusion detection system sensor. Therefore, it is necessary to create a policy that detects a string of at least 4 bytes or more. Second, when a communication string is detected frequently, a large number of detection events are generated, which may cause a false alarm, and the performance of the detection sensor may be reduced, thereby limiting communication traffic in a typical Internet environment. Third, in the PCRE grammar, . (Dot), ∗ (Asterisk) is a special character that matches any string. Because this matching matches all strings in the packet Payload, the PCRE computation consumes a lot of system resources and leaks from the intrusion detection system sensor. Fourth, if the setting value exceeds the detection string length limit of the intrusion detection system sensor, it may cause a problem that it cannot be detected. In addition, long length PCRE matching causes performance load of the intrusion detection system sensor. Fifth, when searching a continuous pattern of the same character, a looping phenomenon may occur as repeated operations are performed, which causes a heavy load on the CPU usage. Therefore, there is a purpose to improve these five problems by optimizing Snort detection grammar (see Table 11).

5. Conclusion

The purpose of this paper is to find a detection rule optimization method for protecting users who use location-based services in mobile information systems and proving the compatibility of detection rules between different intrusion detection systems (IDS/IPS) introduced in each security control center (cybersafety center) based on IDS Snort in order to prepare for new cyberthreats and cyberattacks.

Recent hacking technologies understand cyberattack packet contents in order to detect new cyberthreats that are developing rapidly and present the best intrusion detection rules for network environment. Based on the Snort detection rules, we designed the models and options of the essential detection rules and suggested the most optimized detection rule production standards through understanding and analyzing the wrong policies such as the performance of the detection sensor and the policy that does not consider the false positives. In this paper, we propose an efficient detection and countermeasure of new cyberattacks through the Snort-based detection rule standard requirements. Also, constructing a standardized security management system of the heterogeneous intrusion detection system by maintaining the optimization state by correcting and revising the detection pattern according to the actual situation of each security control center is possible.

This standardization of integrated intrusion detection pattern is expected to establish an efficient operation system of each security control center (cybersafety center) performing security control.

Table 11: Comparison of Snort Detection Rules and Optimization Options.

Detection rule options	Snort	Detection rule optimization grammar selection
Header (24/17)		
Rule Actions (8/2)		
alert	O	O
log	O	X
pass	O	X
activate	O	X
dynamic	O	X
drop	O	O
reject	O	X
sdrop	O	X
Protocols (4/4)		
tcp	O	O
udp	O	O
icmp	O	O
ip	O	O
IP (5/5)		
any	O	O
numeric IP	O	O
numeric IP list	O	O
CIDR	O	O
negation(!)	O	O
Port (4/4)		
any	O	O
static port	O	O
ranges(:)	O	O
negation(!)	O	O
Direction (3/2)		
->	O	O
<-	O	X
bidirectional(<>)	O	O
Option (47/24)		
Meta Data (6/1)		
msg	O	O
reference	O	X
sid	O	X
rev	O	X
classtype	O	X
priority	O	X
Payload Detection (19/12)		
content	O	O
content modifier		
Nocase	O	O
Rawbytes	O	O
Depth	O	O
Offset	O	O
Distance	O	O
Within	O	O
http_client_body	O	X
http_uri	O	X

Table 11: Continued.

Detection rule options	Snort	Detection rule optimization grammar selection
http_header	O	X
http_cookie	O	X
uricontent	O	O
isdataat	O	O
pcre	O	O
byte_test	O	O
byte_jump	O	O
ftpbounce	O	X
asn1	O	X
regex	O	X
Non Payload Detection (20/9)		
fragoffset	O	X
ttl	O	O
tos	O	X
id	O	X
ipopts	O	X
fragbits	O	X
dsize	O	O
flags	O	O
flow	O	O
flowbits	O	O
seq	O	X
ack	O	X
window	O	X
itype	O	O
icode	O	O
icmp_id	O	O
icmp_seq	O	O
rpc	O	X
ip_proto	O	X
sameip	O	X
Thresholding (2/2)		
limit	O	O
threshold	O	O

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Dey, J. Hightower, E. De Lara, and N. Davies, "Location-based services," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 11-12, 2010.

[2] R. Bejtlich, "The practice of network security monitoring: understanding incident detection and response," *No Starch Press*, pp. 2–20, 2013.

[3] J. S. Hong, Y. H. Lim, W. H. Park, and K. H. Kook, "Improved Security Monitoring and Control Using Analysis of Cyber Attack in Small Businesses," *The Journal of Society for e-Business Studies*, vol. 19, no. 4, pp. 195–204, 2014.

[4] W. Park and S. Ahn, "Performance Comparison and Detection Analysis in Snort and Suricata Environment," *Wireless Personal Communications*, vol. 94, no. 2, pp. 241–252, 2016.

[5] T. Xu and Y. Cai, "Feeling-based location privacy protection for location-based services," in *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*, pp. 348–357, ACM, Chicago, Ill, USA, November 2009.

[6] M. Roesch, *Snort: Lightweight Intrusion Detection for Networks*, vol. 229, Stanford Telecommunications Inc, Santa Clara, Calif, USA, 1999.

[7] Y.-H. Kim and W. H. Park, "A study on cyber threat prediction based on intrusion detection event for APT attack detection,"

Multimedia Tools and Applications, vol. 71, no. 2, pp. 685–698, 2014.

[8] M. Roesch, *Snort: Lightweight Intrusion Detection for Networks*, vol. 229, Santa Clara, Calif, USA, Stanford Telecommunications, Inc, 1999.

[9] Z. Zhou, Z. Chen, T. Zhou, and X. Guan, "The study on network intrusion detection system of snort," in *Proceedings of the 2nd International Conference on Networking and Digital Society, lCNDS 2010*, pp. 194–196, Wenzhou, China, May 2010.

[10] M. Norton and D. Roelker, *SNORT 2.0: Hi-Performance Multi-Rule Inspection Engine*, Sourcefire Network Security Inc, 2002.

[11] P. Garcia-Teodoro et al., *Anomaly-Based Network Intrusion Detection: Techniques, Systems and Challenges*, computers & security 28.1, 2009.

[12] G. C. Tjhai, M. Papadaki, S. M. Furnell, and N. L. Clarke, "Investigating the problem of IDS false alarms: An experimental study using Snort," *IFIP International Federation for Information Processing*, vol. 278, pp. 253–267, 2008.

[13] J. D. Rance, *Structured Exception-Handling Methods, Apparatus, and Computer Program Products*, Los Gatos, Calif, USA.

[14] S. Chakrabarti, M. Chakraborty, and I. Mukhopadhyay, "Study of snort-based IDS," in *Proceedings of the International Conference and Workshop on Emerging Trends in Technology 2010, ICWET 2010*, pp. 43–47, ind, February 2010.

[15] B. Caswell, J. Beale, and A. Baker, *Snort IDS and IPS Toolkit*, Syngress, New York, NY, USA, 2007.

[16] D. Burks, *Security Onion: Peel Back the Layers of Your Network in Minutes*, Software Engineering Institute, January 2014.

[17] A. Deuble, *Detecting and Preventing Web Application Attacks with Security Onion*, SANS Institute 4.1, 2012.

[18] P. Wonhyung, *Requirements of Detection Rules in Intrusion Detection System based on SNORT*, Telecommunications Technology Association in South Korea, 2015.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 7356213, 14 pages
https://doi.org/10.1155/2017/7356213

Research Article

Collaborative QoS Prediction for Mobile Service with Data Filtering and SlopeOne Model

Yuyu Yin,[1,2,3] Wenting Xu,[1] Yueshen Xu,[4] He Li,[4] and Lifeng Yu[5]

[1]*School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China*
[2]*Key Laboratory of Complex Systems Modeling and Simulation of Ministry of Education, Hangzhou, Zhejiang 310027, China*
[3]*College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, China*
[4]*School of Software, Xidian University, Xi'an, Shanxi 710071, China*
[5]*Hithink RoyalFlush Information Network Co., Ltd., Hangzhou, Zhejiang, China*

Correspondence should be addressed to Yueshen Xu; ysxu@xidian.edu.cn

Received 25 January 2017; Accepted 21 March 2017; Published 22 June 2017

Academic Editor: Jaegeol Yim

The mobile service is a widely used carrier for mobile applications. With the increase of the number of mobile services, for service recommendation and selection, the nonfunctional properties (also known as quality of service, QoS) become increasingly important. However, in many cases, the number of mobile services invoked by a user is quite limited, which leads to the large number of missing QoS values. In recent years, many prediction algorithms, such as algorithms extended from collaborative filtering (CF), are proposed to predict QoS values. However, the ideas of most existing algorithms are borrowed from the recommender system community, not specific for mobile service. In this paper, we first propose a data filtering-extended SlopeOne model (filtering-based CF), which is based on the characteristics of a mobile service and considers the relation with location. Also, using the data filtering technique in FB-CF and matrix factorization (MF), this paper proposes another model FB-MF (filtering-based MF). We also build an ensemble model, which combines the prediction results of FB-CF model and FB-MF model. We conduct sufficient experiments, and the experimental results demonstrate that our models outperform all compared methods and achieve good results in high data sparsity scenario.

1. Introduction

Since many mobile services have been or being developed as the interfaces to access resources on mobile environment, the number of services increases dramatically. Users often have to select a mobile service from a series of service candidates with similar function. To solve the selection issue, people develop the service recommender system to select services with better QoS (short for *quality of service*). But in mobile service invocation, most users only have invoked quite a few services before, and a large part of QoS values are unknown. To solve this problem, it is urgent to find an effective method to predict QoS values, which has been a research highlight in service computing community.

The collaborative filtering (CF for short) algorithm is widely used for QoS prediction [1, 2]. The idea of CF algorithm is to first identify the similar neighbors of a user or a mobile service and then use the historical QoS values of neighbors to predict the unknown values of the target user or service. It can be seen that the prediction accuracy of CF algorithm largely depends on the identification of similar neighbors. In mobile service recommendation, the accuracy of similar neighbor identification is not so well due to the following reasons:

(1) In similar neighbor identification, there is an assumption that the QoS values are stable and reliable. However, QoS is largely impacted by the mobile network environment in different locations, both in the user side and service side. Due to the instability of mobile network environment, the QoS value is also unstable.

(2) Along with the increase of data sparsity, the similarity computation becomes much less accurate. In high

data sparsity, the number of services invoked by a single user is quite limited, which leads to the even few number of common invoked mobile services by more than one user. Especially in the extreme case that two users do not have any services commonly invoked, there is no chance for any two users being the similar neighbor of the other. So it is difficult to conduct similar neighbor identification with high accuracy in sparse QoS records.

(3) In many cases, we need to select the K most similar neighbors from all neighbor candidates, and the value of K brings a nonnegligible impact on prediction accuracy. The optimal value of K often needs to be determined through a series of experiments and is often different in different datasets.

So we decide to propose new models that can handle the above issues, and our models are based on SlopeOne model. For QoS prediction, the SlopeOne model does not need to identify similar neighbors but directly uses the known QoS records to predict missing values [3]. So the SlopeOne model avoids the issue of similar neighbor identification that happens in CF algorithm. However, the SlopeOne model also has a defect; that is, the model needs to use all of the known QoS records for a missing value prediction. On the one hand, such defect increases the time complexity. On the other hand, it is inevitable to involve noise data, which lowers the prediction accuracy. This paper aims to solve those problems and makes the following contributions:

(1) It proposes a twofold data filtering strategy to filter noise to improve prediction accuracy and lower time complexity, for predicting the QoS values of mobile services. The proposed data filtering strategy is not designed to any specific QoS property of a mobile service but can be used to predict all types of QoS properties.

(2) It proposes two novel prediction models. One is an ensemble model, and the other is a matrix factorization model.

(3) It proposes a linear way to combine the results of the two proposed models, further improving the prediction accuracy.

(4) It conducts sufficient experiments in two real-world datasets, and the experimental results demonstrate the effectiveness of the proposed models. Note that our models only need the QoS records as the input, without the need for any other side information, which brings high feasibility in mobile service invocation scenario.

The rest of the paper is organized as follows: Section 2 discusses the related work. Section 3 presents the framework of our work. Section 4 explains the proposed filtering methods, and Section 5 elaborates the proposed models. Section 6 gives the experimental results, and Section 7 concludes the paper and discusses the future work.

2. Related Work

It is hard for a user to invoke all available mobile services to acquire all QoS values, to select the most suitable one. Thus, QoS prediction is an indispensable task in mobile service selection and recommendation. The collaborative filtering (CF for short) algorithm is widely used in traditional service computing community to predict QoS [1, 4–7].

The CF algorithm is first formally proposed by [8] and has been broadly employed in e-commerce recommender systems [9, 10]. The CF algorithm can be classified into two types, that is, neighbor-based CF and model-based CF. The neighbor-based CF algorithm can be further classified into two categories, that is, user-based CF algorithm and service-based CF algorithm. We take the following prediction task as the example: to predict the QoS value of user u receiving after invoking service j, marking as q_{uj}. The user-based CF algorithm first identifies the similar neighbors of user u with similarity computation, using the historical QoS records [11–13]. Then, the user-based CF algorithm collaboratively uses the historical QoS records of the identified similar neighbors to service j to compute the predicted QoS value q_{uj}. The service-based CF algorithm is similar to the user-based CF algorithm, and the difference is that the first step is to identify the similar neighbors of service j, and the missing value q_{uj} is predicted by collaboratively using the known QoS records of user u to the identified service neighbors [10, 14, 15].

In recent years, several new neighbor-based CF algorithms have been also proposed for QoS prediction in traditional service computing. Sun et al. [16] proposed a new similarity computation method to better identify user neighbors and service neighbors. In detail, the authors normalized the QoS values and computed the similarity based on Euclidean distance. Liu et al. [17] proposed a geographic location-based CF algorithm. They assumed that the users that are located near each other had similar network environment and thus were likely to experience similar QoS. Zheng et al. [18] constructed an ensemble model, which combined the prediction results of user-based CF algorithm and service-based CF algorithm with a predefined parameter.

Another important type of CF algorithm is model-based algorithm, and the idea is to learn the latent features of a user and a service and further learn the relation between the latent features of users and services. The learning process is based on the historical QoS records. The model-based algorithm includes SVM [19], MF (short for *matrix factorization*) [20, 21], Bayesian classifier [13], and latent semantic analysis [22]. The MF model has been verified to be effective and be the first choice in many prediction tasks. He et al. [23] proposed a geographic location-based hierarchical MF model, in which the user-service invocation matrix is partitioned into several local matrices, with K-Means algorithm. The final prediction result is computed as the combination of the results that are achieved using the whole matrix and local matrices, respectively. Xu et al. [24] extended the PMF (probabilistic matrix factorization) with geographical information. In their model, the similar neighbors were identified based on the geographical distance, and the latent feature vector of the

target user was learned together with the feature vectors of similar neighbors.

Lemire and Maclachlan [3] first proposed the SlopeOne model in recommender system community, which was easy to implement and could achieve good performance. Zhang [25] proposed a hybrid model that was the combination of SlopeOne model and item-based CF algorithm. Correspondingly, Wang and Ye [26] proposed a hybrid model as the combination of SlopeOne model and user-based CF algorithm. Mi and Xu [27] first clustered items according to the ratings that the items received, and, in each cluster, the missing ratings were predicted using SlopeOne model.

In service computing, there are not so many works that study the SlopeOne model. In this paper, we employ SlopeOne model as the base to predict QoS values for mobile services, and our proposed models are verified to be effective by sufficient experiments.

3. The Whole Framework

We present the whole framework of this paper in Figure 1, which includes the following components:

(1) User-service invocation matrix: it stores the known historical QoS records, and the large part of missing values are to be predicted.

(2) User similarity matrix: it stores the similarity result of two users, which is computed based on the service invocation records.

(3) Service similarity matrix: it stores the similarity result of two services, which is computed based on the invoked records.

(4) Global filter: it identifies the user neighborhood and service neighborhood based on the similarity.

(5) Local filter: it further identifies a fine-grained neighborhood from the neighborhood that is discovered by the global filter.

(6) FB-CF (filtering-based CF): it is the proposed multimodel combination method, which is composed of three submodels, and can select a suitable submodel to finish the prediction task in different conditions.

(7) FB-MF (filtering-based MF): using the prediction results of FB-CF, this component first fills the missing entries in the user-service invocation matrix and then factorizes the matrix using the MF model.

(8) The ensemble model: it combines the FB-CF model and FB-MF model, to further improve the prediction accuracy.

4. The Proposed Filtering Method

In this section, we present our proposed filtering methods, including global data filtering and local data filtering.

4.1. Global Filtering

4.1.1. The Motivation of Global Filtering. The motivation of global filtering is based on the observation of real-world service invocation data, and we take the response time as the example to explain. First, let us see Table 1. In Table 1, the task is to predict the QoS value after user_1 invokes service_2. Using the basic SlopeOne model, we can get the prediction result as $x = a + (c - b)$. Now let us see Table 2, in which the prediction result is $x = 7 + (0.5 - 0.7) = 6.8$. However, the prediction result is likely to be biased, and such bias should be avoided for a linear prediction model. The analysis is based on the real-world QoS data collected by [28], and more details of this dataset can be found in the experiment section (see Section 6.1). We give the following detailed analysis.

(1) As shown in Figures 2(a) and 2(b), the response time data have a strong aggregation characteristic, and the values of most data are distributed around a limited value range. More than 80% values are less than the average, and more than 90% values are located in the range of the average adding double standard deviations. Thus, the prediction value 6.8 is quite likely to be deviated from the real value of x.

(2) The distribution of QoS values shows clear randomness. Assume that the real value of x is close to the prediction value 6.8; then, the QoS value vectors of user_1 and user_2 should have a stable difference in every dimension. If the deviation of the difference of two QoS value vectors is small, the difference between the two vectors is stable. As shown in Figure 3, we randomly select user A and further select user C. The QoS value vector of user C is similar to that of user A. It can be seen that the QoS value of user A can be smaller or larger than the QoS value of user C, which means there is no stable difference between the QoS values of user A and user C, even though the deviation of difference of user C is the smallest compared to user A. So the prediction value 6.8 is likely to be quite different from the real value of x.

Now let us consider the case in Table 3, in which we can get the prediction value $x = 1 + (0.5 - 0.7) = 0.8$. Based on the aggregation effect shown in Figure 2(a), the probability of x being close to 0.8 is large. So naturally more such local matrices are helpful for linear regression to improve the prediction accuracy.

Based on the above analysis, when the difference among a, b, and c is large (as shown in Table 2), the prediction error is likely to be large. In contrast, when the difference among a, b, and c is small (as shown in Table 3), the prediction error is likely to be small. So in this paper, we use global filtering to enlarge the frequency of the cases like Table 3.

4.1.2. Global Filtering with Similarity Computation. The goal of global filtering is to make the values of a, b, and c close to each other. Some papers claim that the users with close geographical location have similar network environment and thus tend to experience similar QoS [17, 29]. However, in mobile environment, the relation between the network configuration and location is more complex. As shown in Figure 4, we randomly select two users A and B that are close to each other geographically. It can be seen that the

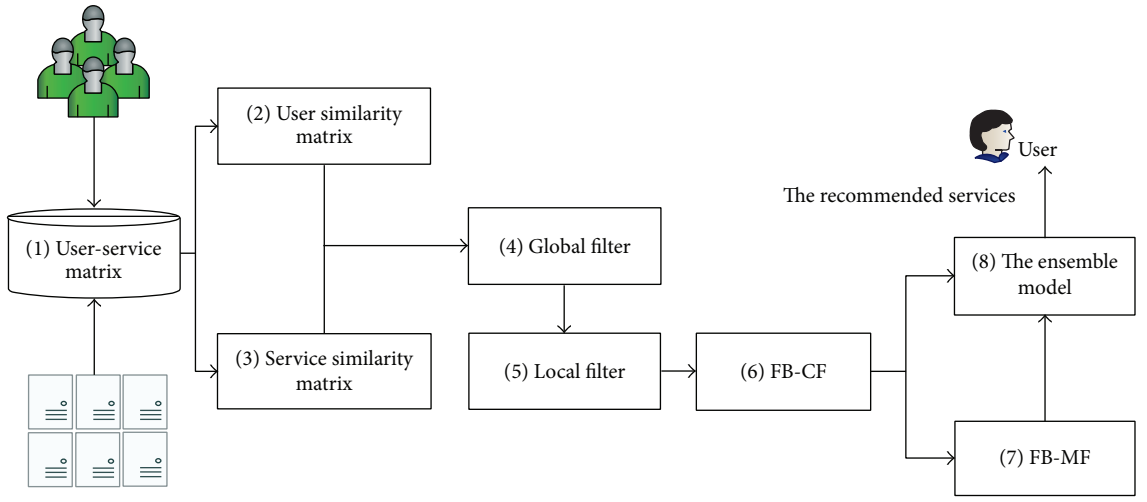


Figure 1: The whole framework.

Table 1: Example 1.

	service_1	service_2
user_1	a	x
user_2	b	c

Table 2: Example 2.

	service_1	service_2
user_1	$a = 7$	$x = ?$
user_2	$b = 0.7$	$c = 0.5$

QoS value (response time) of user A can be quite larger than that of user B or be also quite smaller than that of user B. That is, even though two users locate closely, the QoS values that they receive may still be quite different. Besides, if the side information, such as the geographical location, is indispensable for a model, the applicability of the model will be limited. For a model with geographical information as the input, the model will fail to work in the invocation scenario that has no geographical information. In this paper, the proposed models use Manhattan distance as a base to compute the similarity to finish the global filtering. The Manhattan distance is

$$\text{dis}\left(\vec{x}, \vec{y}\right) = \sum_{i=1}^{n} \left(\left|x_i - y_i\right|\right), \tag{1}$$

where $\vec{x}$ and $\vec{y}$ are the QoS value vectors of two users and n is the number of services that are commonly invoked by the two users. Equation (1) ignores the impact of the number of commonly invoked services, so we use the average Manhattan distance to better compute the similarity of two QoS vectors, which is shown as follows:

$$\text{sim}\left(\vec{x}, \vec{y}\right) = \frac{1}{1 + \sum_{i=1}^{n} \left(\left|x_i - y_i\right|\right)/n}, \tag{2}$$

where we borrow the idea of Laplacian smoothing in the denominator. In (2), if the QoS vectors are closer, the similarity of the users will be larger. Note that although we take the user similarity computation as the example to explain, similarly, the service similarity can be also computed in the same way. In mobile service similarity computation, $\vec{x}$ and $\vec{y}$ are the vectors of the invoked records of two mobile services, and n is the number of users that have commonly invoked the mobile services before.

The global filtering is conducted in both the user side and service side and uses a threshold to control the filtering strength. That is, for the target user or target service, the goal is to select the similar neighbors that the corresponding similarity is larger than the threshold. The threshold is not set manually but computed automatically as

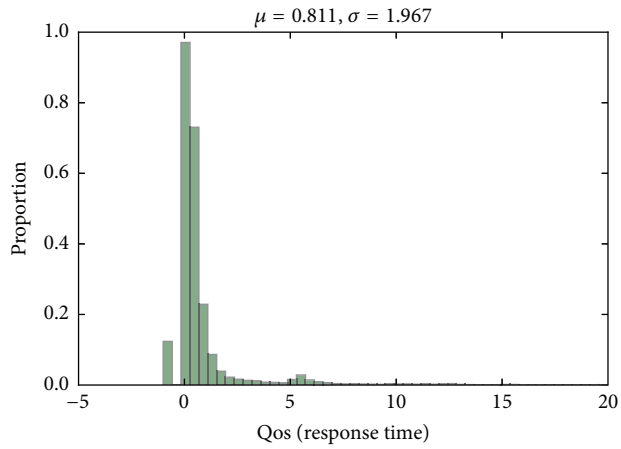
$$\tau_{\text{global}} = \frac{\text{avg}\left(\text{Sim}_{\text{user}}\right) + \text{avg}\left(\text{Sim}_{\text{service}}\right)}{2}, \tag{3}$$

where τ_{global} is the threshold, $\text{avg}(\text{Sim}_{\text{user}})$ is the average value of the user similarity matrix, and $\text{avg}(\text{Sim}_{\text{service}})$ is the average value of the mobile service similarity matrix. The automatic computation of the threshold improves the applicability of our method. The experimental results show that, in two real-world datasets, the proposed global filtering achieves good performance.

After global filtering, we get the similar neighbor set for a user or a service. Considering that, under the case of huge data volume, the similar neighbor set can be quite large, to lower the complexity of subsequent computation, we select the K most similar neighbors to form a compact neighbor set. The sensitivity of our proposed models to K will be given in the experiment section.

4.2. Local Filtering. The global filtering is capable of measuring the closeness of QoS vectors, but there may exist huge difference among some local QoS values. Here is an example, where there are two QoS vectors $\vec{q}_A$ and $\vec{q}_B$:

$$\begin{aligned} \vec{q}_A &= (1, \ldots, 1, 10, 2, \ldots, 2), \\ \vec{q}_B &= (1, \ldots, 1, 3, 2, \ldots, 2). \end{aligned} \tag{4}$$



(a) The distribution of observed QoS values

	QoS value
Minimum/abnormal	−1
Maximum	19.99
Mean	0.811
Median	0.298
Standard deviation	1.967

(b) The statistics of observed QoS values

Figure 2: Distribution of the number of the observed QoS values.

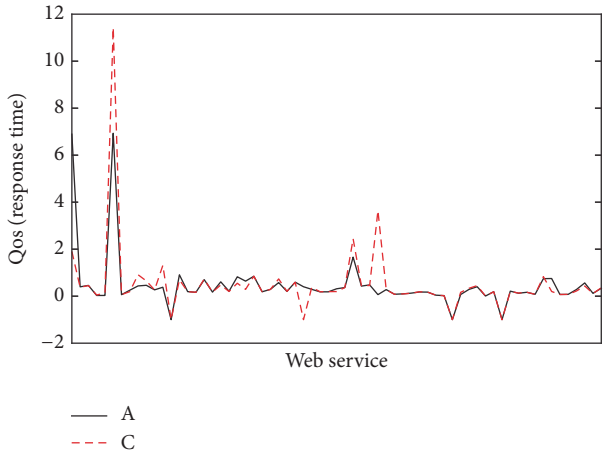


Figure 3: The QoS distribution of vectors with small deviation.

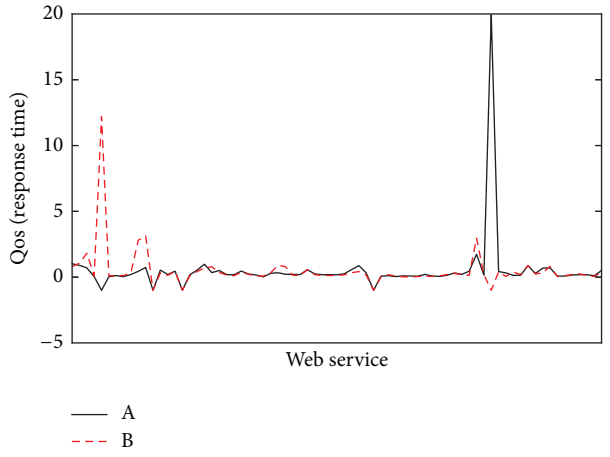


Figure 4: The QoS distribution of users in close geographic location.

Table 3: Example 3.

	$service_1$	$service_2$
$user_1$	$a = 1$	$x = ?$
$user_2$	$b = 0.7$	$c = 0.5$

A and B (A or B could be a user or a service) can receive a quite different QoS value (such as 10 and 3 as shown in the above example), but since the QoS values in other entries are quite similar, overall the similarity should be large. However, in SlopeOne model, using 10 and 3 will lead to large error. So in this paper, we further propose a local filtering method to avoid the above case.

Lemire and Maclachlan [3] proposed the bipolar SlopeOne model, which only uses the data to reach consistency in two-class classification, to be the input of the prediction. This model does the local filtering task to some extent but has the following defects:

(1) It is hard to decide the classification border: as continuous values, the QoS values are different from the traditional rating data, which are discrete values. So it is hard to decide the threshold for two-class classification. For example, we set 5 as the threshold, being larger than 5 is positive class, and being smaller than 5 is negative class. In such a case, being 4.9 will be negative class, and being 5.1 will be positive class, but naturally the two values are quite close to each other.

(2) It is easy to lead to overfiltering: the algorithm requires that, in the local matrix, the classifications of QoS values a, b, and c should be the same. Such strong filtering strategy is likely to lead to too few available data that can be used for prediction, especially in the high data sparsity case.

To solve the above issues, in this paper, we propose a local filtering method based on the dynamic difference classification.

Different from the static two-class classification, which employs a fixed threshold to classify a QoS to one of the two classes, we define that if the difference of two values is smaller

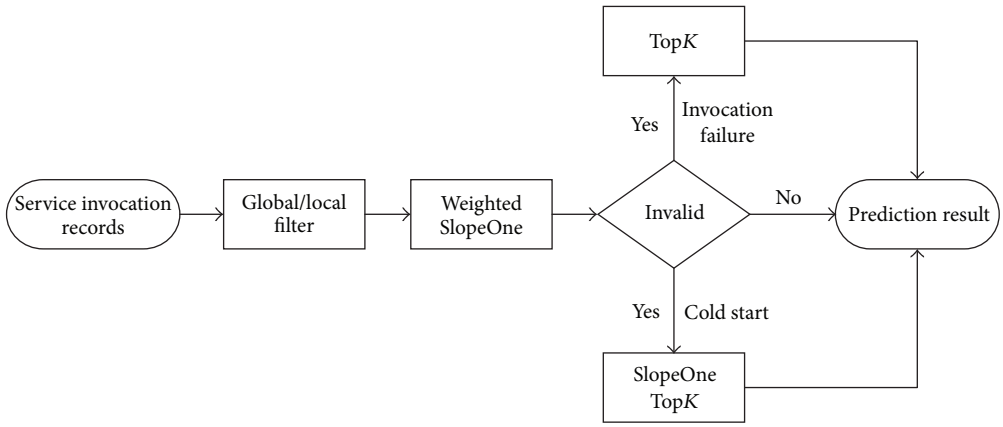


Figure 5: The framework of the FB-CF model.

than a threshold, then the class of the two values is the same. That is,

$$\{q_a, q_b \in \text{the same class} \mid |q_a - q_b| < \tau_{\text{local}}\}, \quad (5)$$

where τ_{local} is the classification threshold. Similar to the global filtering threshold τ_{global}, the local filtering threshold τ_{local} in this section does not rely on manual setting either but is computed automatically as the average of all the known QoS values; that is,

$$\tau_{\text{local}} = \text{avg}(\text{dataset}). \quad (6)$$

5. The Proposed Prediction Models

In this section, we will elaborate the proposed prediction models, including three SlopeOne-based models and one MF-based model.

The framework of the proposed model FB-CF (filtering-based CF) is shown in Figure 5. The first step is to use the proposed global filter and local filter to filter the noise data, and the second step is to conduct the prediction using the proposed weighted SlopeOne model (see the following Section 5.1). In the prediction process, if the weighted SlopeOne model finds that the cases of invocation failure or cold-start occur, the framework will turn to the proposed Top$K$ model or SlopeOne Top$K$ model (see the following Sections 5.2 and 5.3). It means that FB-CF model can select a suitable submodel to fit any real invocation case, which further improves the prediction accuracy. We will give detailed explanation of all models in the rest part of this section.

5.1. Data Filtering-Based Weighted SlopeOne. Assuming two vectors $\vec{v} = \{v_i \mid i = 1, 2, \ldots, n\}$ and $\vec{w} = \{w_i \mid i = 1, 2, \ldots, n\}$, the SlopeOne model uses the linear regression predictor $f(x) = x + b$, $x \in \vec{v}$. If we aim to compute $\vec{w}$ based on $\vec{v}$, there is only one unknown parameter b. To get the parameter b, we only need to minimize the loss function

$$\min \sum_{i=1}^{n} (v_i + b - w_i)^2. \quad (7)$$

So the task turns to computing the optimal b. By derivation, we can get $b = \sum_i (w_i - v_i)/n$, which indicates that b is equal to the average deviation of $\vec{v}$ and $\vec{w}$.

When the SlopeOne model predicts the QoS value of user u invoking mobile service j based on mobile service i, b is the average deviation of the QoS records of service i and service j, and x is the QoS value of user u invoking mobile service i. We can get the following predictor:

$$q_{uj} = q_{ui} + \text{dev}_{ij}, \quad (8)$$

where the average deviation dev_{ij} is computed with

$$\text{dev}_{ij} = \sum_{u \in U_{ij}} \frac{q_{ui} - q_{uj}}{|U_{ij}|}, \quad (9)$$

where U_{ij} represents the user set, in which the users invoke both service i and service j.

5.2. TopK Prediction Model. In the historical invocation records, there exist some QoS values that are not missing but recorded to be negative, which mean that the service invocation fails and the QoS value has not been recorded. In this paper, we also aim to predict the possibility of invocation failure, by studying the possibility of a QoS value being negative. After global and local filtering, if there are some values being negative, which lowers the prediction performance, we use the following Top$K$ model to solve this issue. The user-based Top$K$ model is used to predict the QoS value of user u invoking service j, following the two steps:

(1) Use Top$K$ algorithm to select the similar neighbor set $N(u)$. This step uses Manhattan distance to compute the similarity to select the K most similar neighbors for user u.

(2) Predict the missing values based on similar neighbors' historical QoS records. The predictor is

$$q_{uj} \approx \frac{\sum_{v \in N(u)} \text{sim}(u, v) \times q_{vj}}{\sum_{v \in N(u)} \text{sim}(u, v)}, \quad (10)$$

where $\text{sim}(u, v)$ is the similarity of users u and v.

5.3. SlopeOne-TopK Prediction Model. The cold-start problem is a great challenge in QoS prediction, and we propose another model SlopeOne Top$K$ to solve this problem. We take the example of user u invoking mobile service j to explain the following:

(1) Use Top$K$ algorithm to select the similar neighbor set $N(u)$. This step also uses Manhattan distance to compute the similarity to select the K most similar neighbors for user u. If a neighbor never invoked service j before, we will use the weighted SlopeOne model to first predict the unknown value, to solve the cold-start issue.

(2) Predict the missing values based on similar neighbors' historical QoS records. The predictor is

$$q_{uj} \approx \frac{\sum_{v \in N(u)} \text{sim}(u, v) \times q_{vj}}{\sum_{v \in N(u)} \text{sim}(u, v)}, \tag{11}$$

where q_{vj} is the QoS value of user v invoking mobile service j. If q_{vj} is unknown, we will first predict q_{vj} using weighted SlopeOne model.

5.4. The Proposed FB-MF Prediction Model. In recent years, the MF model and its extensions are widely used in service recommendation system and have been verified to be effective [16]. In MF model, the user-service matrix $R \in \mathbb{R}^{m \times n}$ is factorized into two low-dimensional matrices $P \in \mathbb{R}^{f \times m}$ and $S \in \mathbb{R}^{f \times n}$, as follows:

$$R = P^T S, \tag{12}$$

where m is the number of users, n is the number of mobile services, and f is the number of latent features. So the missing value of user u invoking service j is shown as follows:

$$q_{uj} \approx P_u \cdot S_j. \tag{13}$$

By minimizing the following loss function, we can get the objective function of MF model:

$$L = \frac{1}{2} \sum_{u=1}^{m} \sum_{j=1}^{n} \left(q_{uj} - P_u \cdot S_j\right)^2 + \frac{\lambda}{2} \left(\|P_u\|^2 + \|S_j\|^2\right), \tag{14}$$

where q_{uj} is the real value of user u invoking service s. We use the regularization terms $\|P_u\|^2 + \|S_j\|^2$ to avoid the overfitting problem. We use the gradient descent algorithm to achieve the local optima of the above loss function; the derivatives are

$$\begin{aligned} \frac{\partial L}{\partial P_u} &= \left(P_u \cdot S_j - q_{uj}\right) \cdot S_j + \lambda P_u, \\ \frac{\partial L}{\partial S_j} &= \left(P_u \cdot S_j - q_{uj}\right) \cdot P_u + \lambda S_j. \end{aligned} \tag{15}$$

In fact, the user-service matrix R is quite sparse. So in the process of minimizing the loss function, there are many q_{uj} being missing. Such high sparsity seriously impacts the effectiveness of the model and decreases the prediction accuracy. So we propose a filtering-based MF model (FB-MF for short) to solve the problem.

In FB-MF model, we first use the FB-CF model to finish the prediction task and fill the missing value in user-service matrix R. So in the beginning of FB-MF model, all values are known. Since the prediction result of FB-CF is close to the real value, the existing prediction result can be the base of FB-MF model, to further improve the prediction accuracy.

5.5. The Ensemble Model. Note that the FB-CF model is a local prediction model that uses the filtered local data from the whole QoS records. In contrast, the FB-MF model is a global model that uses the whole QoS records. To further improve the prediction accuracy, we combine the prediction results of the FB-CF model and FB-MF model. We use a parameter to combine the two results linearly, which is shown as follows:

$$q_{uj} \approx \theta \times q_{\text{FB-CF}} + (1 - \theta) \times q_{\text{FB-MF}}. \tag{16}$$

The parameter θ is used to control the weight of two individual models in the final prediction result. If the parameter θ is set to 0, the ensemble model will be degraded to the FB-MF model. If θ is set to 1, the ensemble model will be degraded to the FB-CF model. We name the ensemble model as filtering-based ensemble model (FB-EM for short).

Although, in the current paper, we adopt a static way (θ) to control the weight of the two models, we can see from the experimental results of parameter sensitivity that our model is not sensitive to the value of θ. It indicates that the static setting of θ does not bring much impact on the model performance. We will add the task of dynamic parameter setting into the future work list.

6. Experiment and Evaluation

We conduct sufficient experiments to evaluate the performance of our proposed models, compared to several well-known existing models. The experimental results demonstrate that our models achieve better prediction accuracy and are also not sensitive to the parameters.

6.1. Dataset and Evaluation Metrics. In the experiments, we use a real-world service QoS dataset, WSDream dataset, which is published by [28]. This dataset contains 5825 services and 339 users and contains two types of QoS attributes, that is, response time and throughput. In this paper, we conduct experiments on both response time and throughput records. This dataset has been widely employed to evaluate the prediction accuracy by many researchers [18, 24, 30, 31]. So the experimental results in this paper are convincing.

We use the Mean Absolute Error (MAE) and Normalized Mean Absolute Error (NMAE) to measure the prediction accuracy of our models. The MAE is defined as follows:

$$\text{MAE} = \frac{1}{N} \sum_{u,s} \left|q_{u,s} - \widehat{q}_{u,s}\right|. \tag{17}$$

Table 4: Accuracy comparison (a smaller value means higher accuracy).

Model	Training set density (TD) — response time							
	TD = 5%		TD = 10%		TD = 15%		TD = 20%	
	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE
UserMean	0.8829	1.0879	0.8766	1.0861	0.8785	1.0861	0.8749	1.0849
ItemMean	0.7318	0.9019	0.7121	0.8823	0.7101	0.8779	0.7031	0.8718
SlopeOne	0.7126	0.8757	0.6929	0.8560	0.6886	0.8544	0.6843	0.8492
IPCC	0.6833	0.8422	0.6248	0.7741	0.6037	0.7464	0.5849	0.7253
UPCC	0.6763	0.8335	0.6304	0.7811	0.6172	0.7630	0.6053	0.7506
WSRec	0.6542	0.8063	0.6157	0.7628	0.6026	0.7450	0.5825	0.7223
MF	0.6441	0.7939	0.5405	0.6697	0.5207	0.6438	0.5023	0.6229
FB-CF	*0.4991*	*0.6150*	*0.4282*	*0.5305*	*0.3977*	*0.4917*	*0.3788*	*0.4697*
FB-MF	*0.4890*	*0.6027*	*0.4269*	*0.5289*	*0.4019*	*0.4968*	*0.3864*	*0.4791*
FB-EM	*0.4856*	*0.5984*	*0.4195*	*0.5196*	*0.3925*	*0.4852*	*0.3751*	*0.4651*

The NMAE is defined as follows:

$$\text{NMAE} = \frac{\text{MAE}}{\sum_{u,s} q_{u,s}/N}, \tag{18}$$

where $q_{u,s}$ is the real QoS value in testing set, $\widehat{q}_{u,s}$ is the prediction value, and N is the number of QoS values in testing set. A smaller MAE value or a smaller NMAE value means higher prediction accuracy.

6.2. Experiment Setting. In the real-world service invocation, the number of known user-service invocation records is quite limited. To conduct the experiment in a real-world scenario, we randomly select a small part of QoS records from the whole dataset to generate the training set, and the remaining data generate the testing set. In our experiment, we evaluate the prediction accuracy of each model on four different training set densities, that is, 5%, 10%, 15%, and 20%. For example, in the case of training set density being 5%, it means that 5% of the whole data form the training set, while the remaining 95% data are to be predicted. Each set of experiment is conducted for 10 times, and we report the average result. We conduct experiments on both response time and throughput datasets, to give people the confidence that our models can be employed in diverse QoS prediction tasks for mobile service.

In parameter setting, for the FB-CF model, we set the parameter K, including the size of user neighborhood, to be 10 (marked as $K_{\text{user}} = 10$) and the size of service neighborhood, to be 30 (marked as $K_{\text{service}} = 30$). For the FB-MF model, the number of latent factors f is set to be 50, and the regularization parameter λ is set to be 0.01. For the hybrid model, the parameter θ is set to be 0.6. All parameters in the baseline models are set to the same values as in their original papers.

6.3. Performance Comparison. To evaluate the prediction accuracy of our models, we implement several well-known QoS prediction models, as listed below. In those models, UPCC, IPCC, and WSRec are neighborhood-based models, MF is model-based, and SlopeOne is a regression-based model:

(1) UserMean: the missing QoS value is predicted as the mean of the historical QoS values invoking by the target user.

(2) ItemMean: the missing QoS value is predicted as the mean of the historical QoS values on the target service invoking by different users.

(3) UPCC (user-based PCC): UPCC is a user-based collaborative filtering method. This method utilizes the historical QoS records of similar users to predict the missing QoS values in a collaborative way [32].

(4) IPCC (item-based PCC): IPCC is an item-based collaborative filtering model. This method utilizes the historical QoS records of similar services to predict the missing QoS value [15].

(5) WSRec: this method is proposed by [18] and linearly combines the prediction results of UPCC and IPCC. WSRec uses a parameter to balance the weighted UPCC and IPCC.

(6) MF: MF refers to the matrix factorization model and has been explained in Section 5.4.

(7) SlopeOne: SlopeOne is a linear regression model proposed by [3].

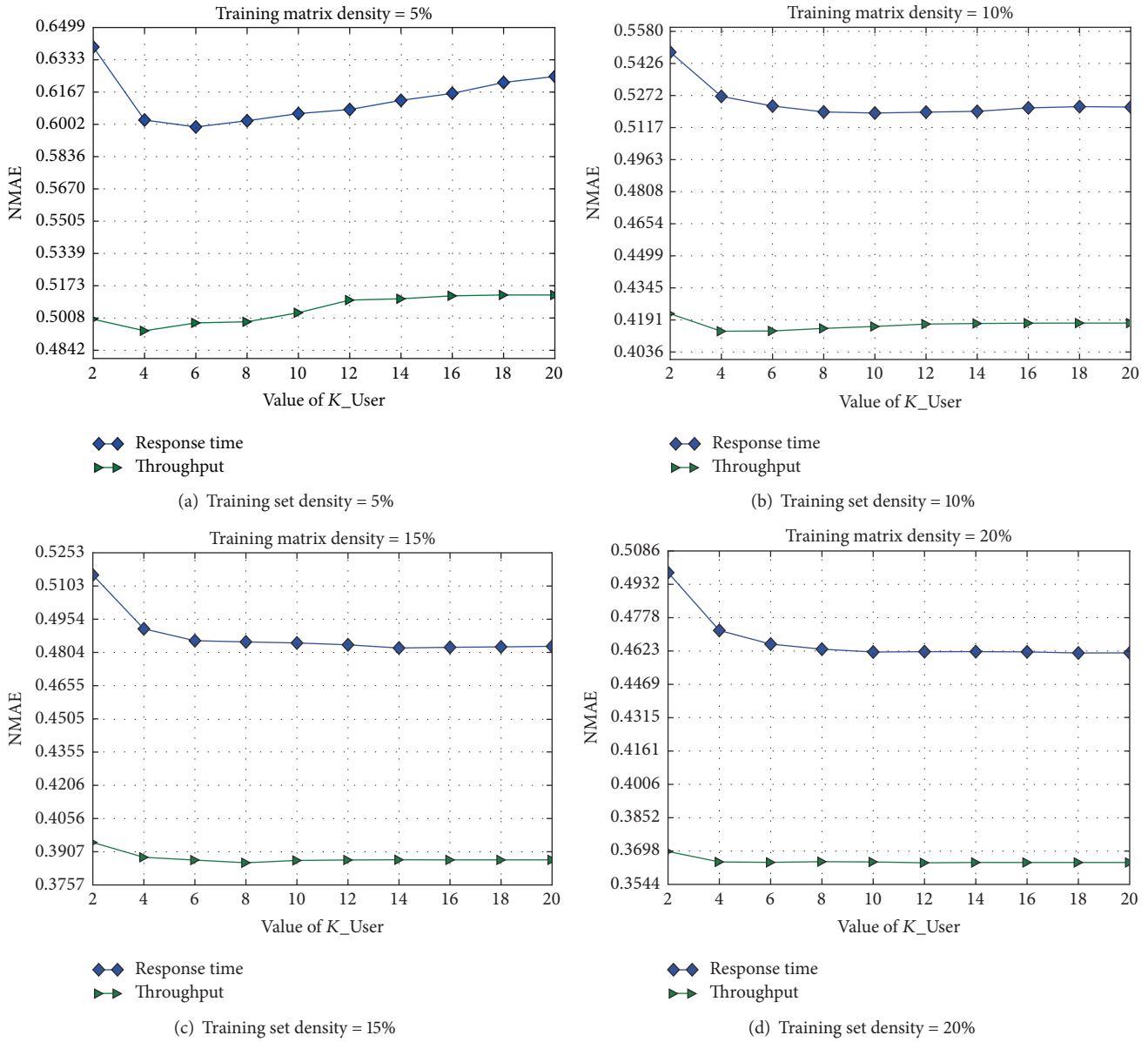
From both Tables 4 and 5, we have the following observations:

(1) The proposed three models (FB-CF, FB-MF, and FB-EM) all achieve higher prediction accuracy than other baseline models in both datasets and in various density cases. Such an improvement indicates that the proposed filtering strategies, combination model, and the ensemble model are effective. Also, it can be inferred that our proposed filtering strategies and models have high feasibility to different data densities. The reason that the FB-MF model performs better than FB-CF model is as follows:

   (a) In the initial state of FB-MF model, the sparse user-service matrix is prefilled using the prediction result of FB-CF model. So it can be seen that the prediction procedure of FB-MF model is exactly built on the achieved prediction result. So expectably, the prediction result of FB-MF model should be better than the result of FB-CF model.

Table 5: Accuracy comparison (a smaller value means higher accuracy).

Model	Training set density (TD) — throughput							
	TD = 5%		TD = 10%		TD = 15%		TD = 20%	
	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE
UserMean	50.937	1.1729	51.343	1.1684	50.941	1.1676	51.185	1.1639
ItemMean	37.307	0.8597	33.014	0.7508	32.785	0.7490	32.676	0.7421
SlopeOne	31.798	0.7322	31.820	0.7242	31.688	0.7263	31.701	0.7208
IPCC	31.112	0.7164	29.936	0.6813	30.100	0.6899	30.609	0.6960
UPCC	30.829	0.7099	29.054	0.6612	28.357	0.6499	28.114	0.6393
WSRec	29.538	0.6802	28.185	0.6414	27.556	0.6315	27.422	0.6235
MF	58.623	1.3503	30.188	0.6870	24.106	0.5525	22.065	0.5017
FB-CF	*24.189*	*0.5568*	*19.396*	*0.4413*	*17.522*	*0.4015*	*16.192*	*0.3682*
FB-MF	*21.870*	*0.5035*	*18.597*	*0.4231*	*17.533*	*0.4018*	*16.940*	*0.3852*
FB-EM	*21.806*	*0.5020*	*18.151*	*0.4130*	*16.884*	*0.3869*	*16.036*	*0.3646*



(a) Training set density = 5%

(b) Training set density = 10%

(c) Training set density = 15%

(d) Training set density = 20%

Figure 6: Sensitivity to K (user).

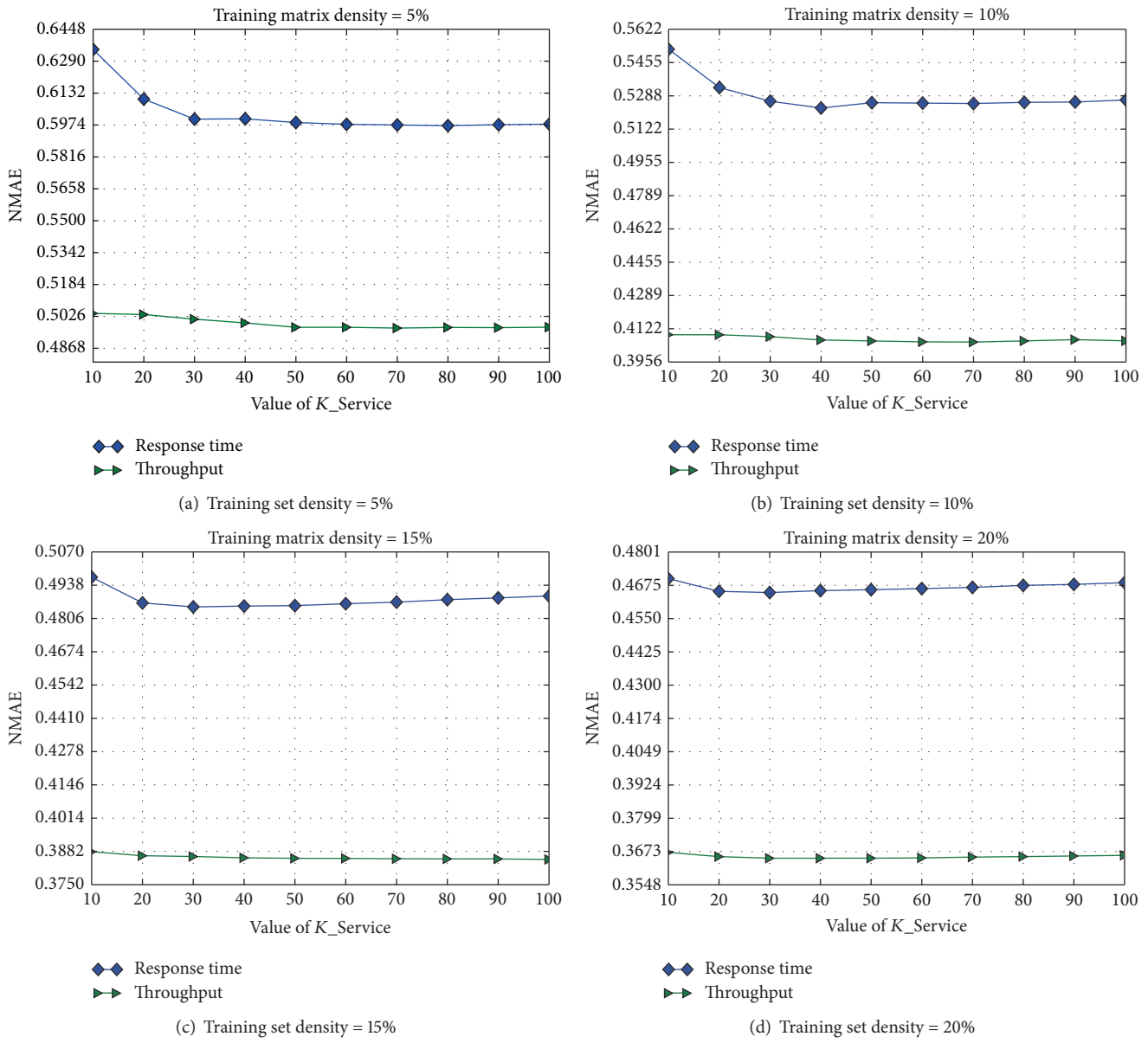


FIGURE 7: Sensitivity to K (service).

(b) We can notice that, in Tables 4 and 5, the performance of MF model is consistently better than that of collaborative filtering algorithms (e.g., IPCC and UPCC). It indicates that the MF model itself has larger potential to achieve higher prediction accuracy.

(2) Along with the training set density increasing, MAE and NMAE values decrease. It indicates that more historical invocation records indeed can improve the prediction performance.

(3) Based on the paired t-tests ($p < 0.001$), the improvements achieved by our three models are all significant.

In the rest part of this section, we will study the sensitivity of our proposed ensemble model FB-EM to the parameters.

6.4. The Sensitivity Analysis of K. In this paper, we use the parameter K to control the number of user or service neighborhood size. Using K lowers the time complexity and saves the time of online prediction. We find that the change trends of MAE and NMAE are quite similar, so we report the result of NMAE here.

The parameter K_{user} controls the number of user neighborhood, and as Figure 6 shows, with the increase of K_{user}, the NMAE value first decreases and then reaches a stable point. At the point of K_{user} being equal to 10, the model achieves the best NMAE value. So we set the default parameter of K_{user} to 10. Note that, in the two datasets of response time and throughput, the change trends of K_{user} and NMAE are quite similar, which illustrates that our model can be used in different prediction tasks.

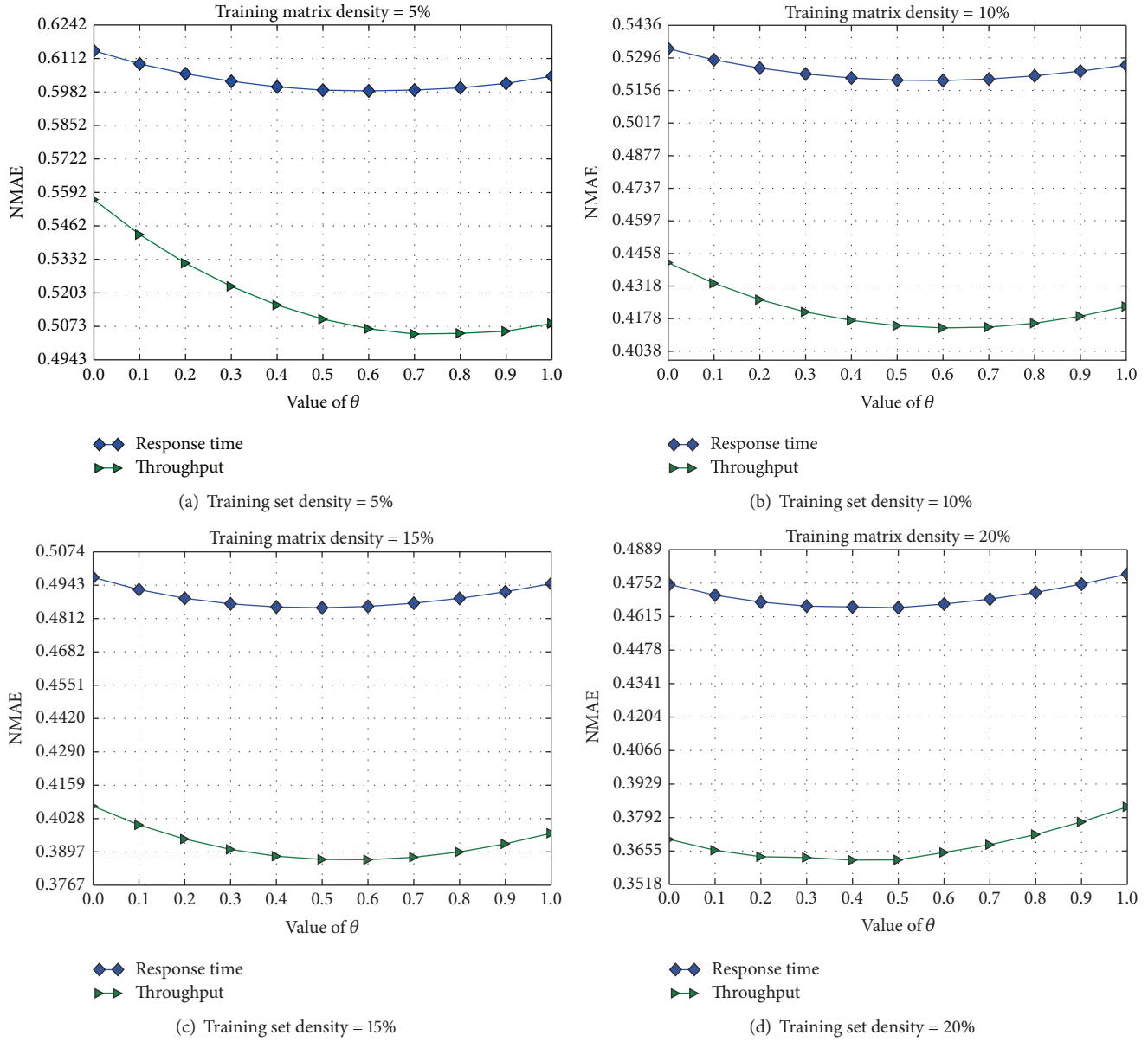


Figure 8: Sensitivity to θ.

The parameter of K_{service} controls the number of service neighborhoods, and as Figure 7 shows, with the increase of K_{service}, the NMAE value first decreases and then also becomes stable at the point of K_{service} being 30, where the model achieves the best NMAE value. So we set the default parameter of K_{service} to 30. Similarly, in the two datasets, the change trends of K_{service} and NMAE are also quite similar.

6.5. The Sensitivity Analysis of θ. The parameter θ is used to balance the weight of two individual models (FB-CF and FB-MF) in the ensemble model. We set the parameter θ in the range of 0 to 1. We report the experimental result in both response time dataset and throughput dataset, in Figure 8.

It can be seen that, in four different training set densities, the optimal value of θ is all in the value of 0.5~0.7. In the whole range of 0 to 1, the change extent of NMAE value is limited, and in the two datasets, the change trends of NMAE are also quite similar. For one thing, it indicates that our model is not sensitive to the setting of θ. For another thing, our model can be used for multiple QoS prediction tasks.

6.6. The Sensitivity Analysis of Training Set Density. The training set density is the proportion of known mobile service invocation records in the whole dataset. A higher training set density means more information can be used for QoS prediction. To better study the impact of training set density, we conduct comparative experiments on three different values of K_{user} (5, 10, and 15) and three different values of K_{service} (20, 30, and 40). The experimental results

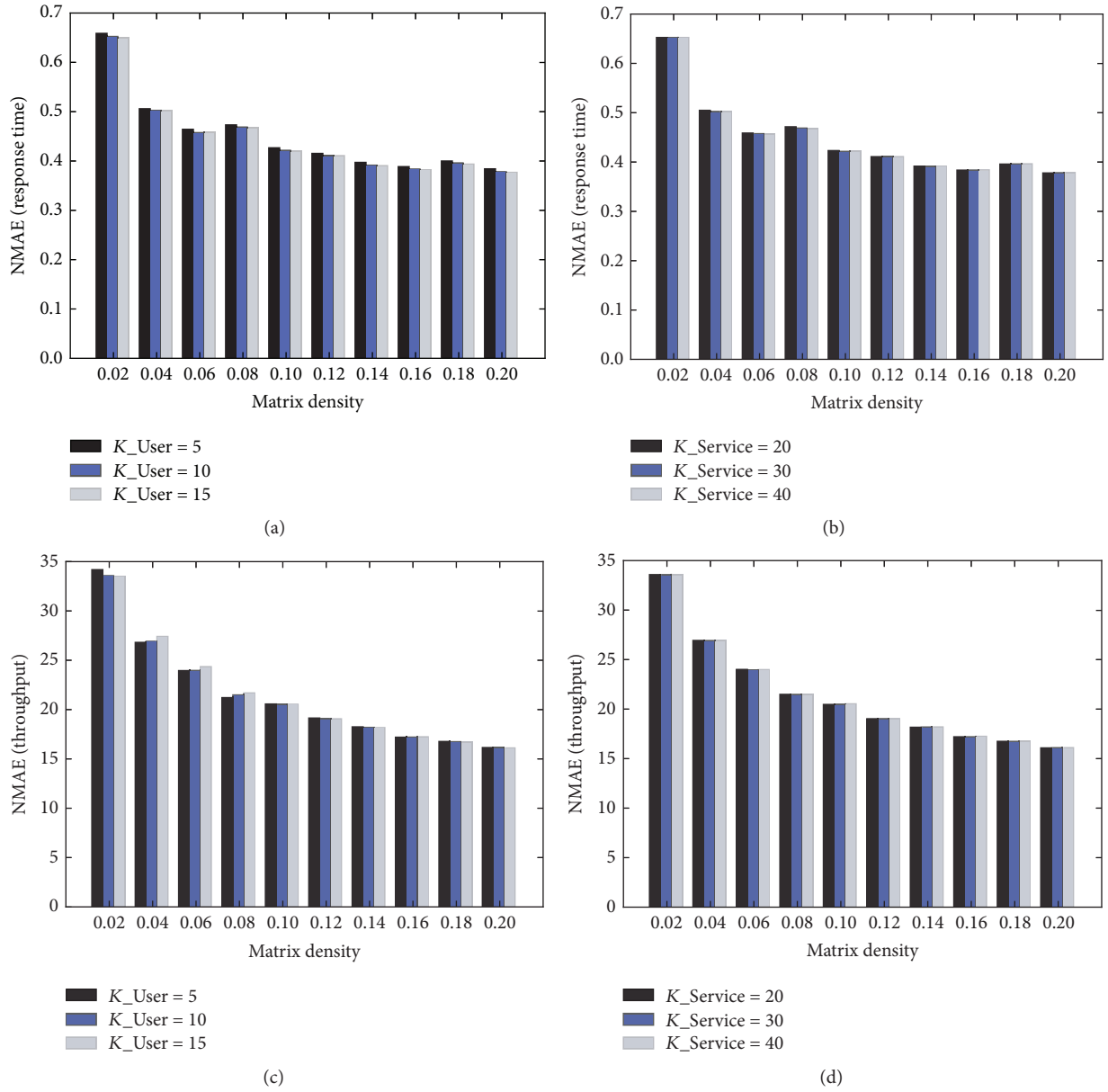


Figure 9: Sensitivity to training set density.

are shown in Figure 9, where the density is set to be the value in the range of 2% to 20%.

Figure 9 shows that, with the matrix density increasing, the NMAE value decreases at first. Along with the training set density being larger, the speed of decreasing becomes slower. It means that when there are only limited historical invocation records, the best way to improve prediction accuracy is to collect more QoS data. But when the number of QoS records becomes larger, the key of the prediction task turns to the development of effective models.

7. Conclusion and Future Work

In this paper, we propose two filtering-based models to predict QoS values for mobile services and an ensemble model, which are FB-CF (filtering-based CF), FB-MF (filtering-based MF), and FB-EM (filtering-based ensemble model). The proposed three models are all based on the proposed filtering methods. The FB-CF model and FB-MF model are extended from SlopeOne model and matrix factorization, respectively. We propose two filtering methods, that is, global filtering and local filtering. The goal of the filtering methods is to filter the noise data that are not suitable for similarity computation. In particular, the FB-CF model and the filtering methods are organized into a unified framework. We conduct sufficient experiments on a real-world dataset, and the experimental results demonstrate the effectiveness of our filtering methods and models.

In the future, we will continue to improve our model from various ways. For example, we plan to use a more flexible

way to combine the two individual models, instead of using a fixed parameter. Second, we also try to improve the filtering methods by investigating more QoS properties of mobile services.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Yueshen Xu and Yuyu Yin contributed equally to this paper and they are co-first authors.

Acknowledgments

This paper is funded by Zhejiang Provincial Natural Science Foundation (no. LY12F02003), China Postdoctoral Science Foundation (no. 2013M540492), the National Key Technology R&D Program (no. 2015BAH17F02), and the National Natural Science Fund of China (nos. 61100043, 61173177).

References

[1] X. Chen, X. Liu, Z. Huang, and H. Sun, "RegionKNN: a scalable hybrid collaborative filtering algorithm for personalized web service recommendation," in *Proceedings of the IEEE 8th International Conference on Web Services (ICWS '10)*, pp. 9–16, IEEE, July 2010.

[2] J. Yin, W. Lo, S. Deng, Y. Li, Z. Wu, and N. Xiong, "Colbar: a collaborative location-based regularization framework for QoS prediction," *Information Sciences*, vol. 265, pp. 68–84, 2014.

[3] D. Lemire and A. Maclachlan, "Slope one predictors for online rating-based collaborative filtering," in *Proceeding of the SIAM International Conference on Data Mining (SDM)*, vol. 5, pp. 1–5, SIAM, 2005.

[4] R. Burke, "Hybrid recommender systems: survey and experiments," *User Modelling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002.

[5] C. Zhang, L. Zhang, and G. Zhang, "QoS-aware mobile service selection algorithm," *Mobile Information Systems*, vol. 2016, Article ID 4968279, 6 pages, 2016.

[6] L. Qi, X. Xu, W. D. Dou, J. Yu, Z. Z. Zhou, and X. Zhang, "Time-aware IoE service recommendation on sparse data," *Mobile Information Systems*, vol. 2016, Article ID 4397061, 12 pages, 2016.

[7] J. Yin, X. Lu, C. Pu, Z. Wu, and H. Chen, "JTangCSB: a cloud service bus for cloud and enterprise application integration," *IEEE Internet Computing*, vol. 19, no. 1, pp. 35–43, 2015.

[8] E. Rich, "User modeling via stereotypes," *Cognitive Science*, vol. 3, no. 4, pp. 329–354, 1979.

[9] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," in *Proceeding of the 16th International World Wide Web Conference (WWW '07)*, pp. 271–280, Alberta, Canada, May 2007.

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

[11] J. L. Herlocker, J. A. Konstan, Al. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 230–237, 1999.

[12] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, "Grouplens: applying collaborative filtering to usenet news," *Communications of the ACM*, vol. 40, no. 3, pp. 77–87, 1997.

[13] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI '98)*, pp. 43–52, Morgan Kaufmann Publishers Inc., 1998.

[14] M. Deshpande and G. Karypis, "Item-based top-N recommendation algorithms," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143–177, 2004.

[15] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th International Conference on World Wide Web (WWW '01)*, pp. 285–295, 2001.

[16] H. Sun, Z. Zheng, J. Chen, and M. R. Lyu, "Personalized web service recommendation via normal recovery collaborative filtering," *IEEE Transactions on Services Computing*, vol. 6, no. 4, pp. 573–579, 2013.

[17] J. Liu, M. Tang, Z. Zheng, X. Liu, and S. Lyu, "Location-Aware and Personalized Collaborative Filtering for Web Service Recommendation," *IEEE Transactions on Services Computing*, vol. 9, no. 3, pp. 686–699, 2016.

[18] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "WSRec: a collaborative filtering based web service recommender system," in *Proceedings of the IEEE International Conference on Web Services (ICWS '09)*, pp. 437–444, IEEE, July 2009.

[19] M. Grcar, B. Fortuna, D. Mladenic, and M. Grobelnik, "knn versus svm in the collaborative filtering framework," in *Data Science and Classification*, pp. 251–260, Springer, 2006.

[20] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain Monte Carlo," in *Proceedings of the 25th International Conference on Machine Learning (ICML '08)*, pp. 880–887, ACM, Helsinki, Finland, July 2008.

[21] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, pp. 713–719, August 2005.

[22] T. Hofmann, "Collaborative filtering via gaussian probabilistic latent semantic analysis," in *Proceedings of the 26th International ACM SIGIR Conference on Research and Development in Informaion Retrieval (SIGIR)*, pp. 259–266, Toronto, Canada, 2003.

[23] P. He, J. Zhu, Z. Zheng, J. Xu, and M. R. Lyu, "Location-based hierarchical matrix factorization for Web service recommendation," in *Proceedings of the 21st IEEE International Conference on Web Services, (ICWS '14)*, pp. 297–304, 2014.

[24] Y. Xu, J. Yin, W. Lo, and Z. Wu, "Personalized location-aware QoS prediction for web services using probabilistic matrix factorization," in *Proceedings of the Web Information Systems Engineering—(WISE '13)*, Lecture Notes in Computer Science, pp. 229–242, Springer.

[25] D. Zhang, "An item-based collaborative filtering recommendation algorithm using slope one scheme smoothing," in *Proceedings of the 2nd International Symposium on Electronic Commerce and Security, (ISECS '09)*, vol. 2, pp. 215–217, May 2009.

[26] P. Wang and H. W. Ye, "A personalized recommendation algorithm combining slope one scheme and user based collaborative

filtering," in *Proceedings of the International Conference on Industrial and Information Systems, (IIS '09)*, pp. 152–154, April 2009.

[27] Z. Mi and C. Xu, "A recommendation algorithm combining clustering method and slope one scheme," in *Proceedings of the International Conference on Intelligent Computing*, pp. 160–167, Springer, 2011.

[28] Z. Zheng, Y. Zhang, and M. R. Lyu, "Distributed QoS evaluation for real-world Web services," in *Proceedings of the IEEE 8th International Conference on Web Services (ICWS '10)*, pp. 83–90, Miami, Fla, USA, July 2010.

[29] W. Lo, J. Yin, S. Deng, Y. Li, and Z. Wu, "Collaborative web service QoS prediction with location-based regularization," in *Proceedings of the IEEE 19th International Conference on Web Services (ICWS '12)*, pp. 464–471, Honolulu, Hawaii, USA, June 2012.

[30] D. Yu, Y. Liu, Y. Xu, and Y. Yin, "Personalized QoS prediction for web services using latent factor models," in *Proceedings of the 11th IEEE International Conference on Services Computing, (SCC '14)*, pp. 107–114, July 2014.

[31] Q. Yu, Z. Zheng, and H. Wang, "Trace norm regularized matrix factorization for service recommendation," in *Proceedings of the IEEE 20th International Conference on Web Services, (ICWS '13)*, pp. 34–41, July 2013.

[32] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filteringof netnews," in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pp. 175–186, Chapel Hill, NC, USA, 1994.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 2189646, 11 pages
https://doi.org/10.1155/2017/2189646

Research Article

A Hybrid Location Privacy Solution for Mobile LBS

Ruchika Gupta and Udai Pratap Rao

Department of Computer Engineering, National Institute of Technology, Surat, Gujarat 395007, India

Correspondence should be addressed to Ruchika Gupta; rgupt009@gmail.com

Received 9 December 2016; Revised 2 March 2017; Accepted 8 March 2017; Published 18 June 2017

Academic Editor: Jaegeol Yim

The prevalent usage of location based services, where getting any service is solely based on the user's current location, has raised an extreme concern over location privacy of the user. Generalized approaches dealing with location privacy, referred to as cloaking and obfuscation, are mainly based on a trusted third party, in which all the data remain available at a central server and thus complete knowledge of the query exists at the central node. This is the major limitation of such approaches; on the other hand, in trusted third-party-free framework clients collaborate with each other and freely communicate with the service provider without any third-party involvement. Measuring and evaluating trust among peers is a crucial aspect in trusted third-party-free framework. This paper exploits the merits and mitigating the shortcomings of both of these approaches. We propose a hybrid solution, HYB, to achieve location privacy for the mobile users who use location services frequently. The proposed HYB scheme is based on the collaborative preprocessing of location data and utilizes the benefits of homomorphic encryption technique. Location privacy is achieved at two levels, namely, at the *proximity* level and at *distant* level. The proposed HYB solution preserves the user's location privacy effectively under specific, pull-based, sporadic query scenario.

1. Introduction

The intense development of location detection empowered devices and escalated availability of wireless interconnections almost everywhere results in emerging location based applications. In Location Based Services (LBS), we incline to use positioning technology to register mobile location movement. There are quite a lot of abstract approaches and real implementations of systems to resolve the place of a cell phone. The most outstanding example of such a positioning system is the GPS [1, 2]. Although LBS offer major openings for a large variety of markets and remarkable convenience to the end user, it also presents subtle privacy attacks at the same time. Privacy of the system is threatened due to the requirement of the current location of the user in order to provide related services.

As per the connotation, LBS (i.e., services based on location) needs user's exact location coordinates to supply accurate service support to the user. Centralized architecture and decentralized architecture, also referred to as trusted third party (TTP) based and TTP-free architectures, respectively, are two basic frameworks existing to preserve location privacy of the user in LBS. An adversary with the adequate accessibility to user's data may use the location information for a particular motive and may also keep it to perform the linkages with publicly available data for detailed profiling of the user [3]. LBS may also use such data for business promotions through advertising. The series of submitted location with query from a specific place can disclose too much about a person. The scenario can become extremely unpleasant if the adversary gets access to the user's sequence of location data with attached timestamps. For example, first visit of Alice to an attorney's office speaks less about her but few days later, her subsequent visit to the court reveals altogether a different story. Location revelation by Alice to LBS provider discloses some extremely private affairs of her life through inference attacks which were not apparent otherwise [4].

The query "Find my nearest attorney's office" by Alice can directly be answered by a location server such as Google

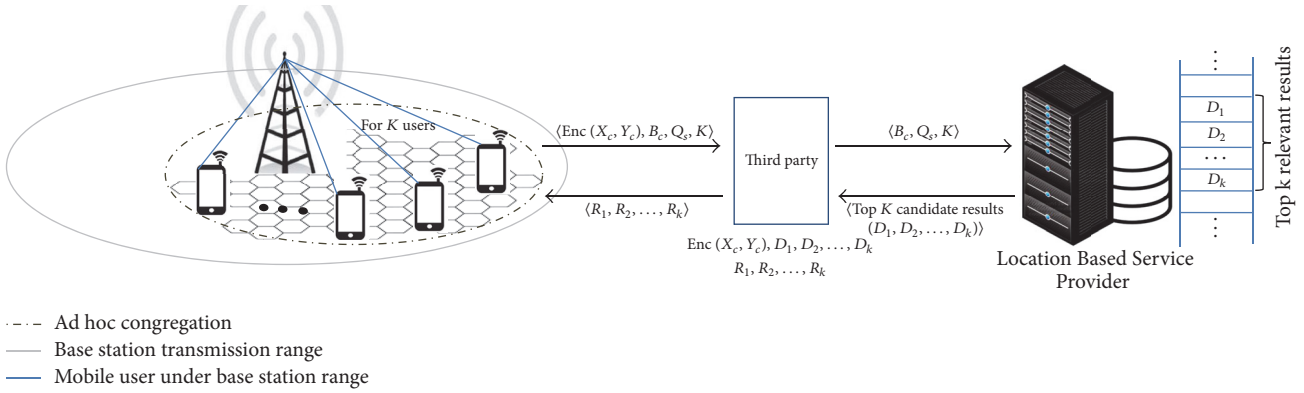


Figure 1: System model.

maps, bing maps, and map quest but the connection to these servers are not trusted. Therefore, instead in order to protect privacy, Alice sends her query via a TTP (also called anonymizer) that strips off her identification information, generates the blurred location data, and mediates the communication between her and LBS provider [5, 6]. However, the query submitted by Alice to TTP still has her actual location coordinates; hence malicious user having control over TTP can have complete information about the user. Thus it is always risky to use TTP based framework to connect to the LBS server. Trusting the third party is the prime downside of the TTP based mechanisms. If a user can trust a third party for small functionality then why not the service provider for bigger benefits, can always be argued. In distributed peer approach, mobile clients are equipped to connect with other mobile users as and when required. The development of distributed wireless communication technologies, such as WLAN IEEE 802.11, Bluetooth IEEE 802.15.1, and ZigBee (for low energy devices) IEEE 802.15.4-based specifications, combined with the propelled computing potential and memory capacity of today's mobile devices become useful to bring privacy preserving benefits to the user. This way the need to rely solely on the connection to the server is eliminated. In TTP-free architecture, all functions are supposed to be carried out at the user's handheld and thus make the communication heavier and more time consuming. Efficiency of decentralized architecture also depends upon the computing capability of used mobile device. However, peers' trust measure and evaluation is another big concern.

Figure 1 presents the proposed architecture of hybrid model. Here, it is presumed that there are a substantial number of mobile users carrying handheld devices such as cell phones, PDAs, or the like which are equipped with positioning capabilities and use location services frequently. The handhelds have computation power, processing potential, memory, and required access to the wireless network. All the users are in the transmission range of the base station (or beacon node).

In the proposed hybrid model, we suggest that the mobile user querying LBS first forms an ad hoc congregation with other users exploiting the well-established principle of $\mathcal{K}$-anonymity. Once the congregation is formed, centroid is calculated in such a way that participating users' locations are not revealed. The centroid coordinates are then secured using encryption and sent to the third party (TP). Query ($\mathcal{Q}$) includes secured location coordinates, nearest base station information, anonymity parameter $\mathcal{K}$, and the query string. TP strips off the encrypted data and without performing any changes forwards the rest of the query to the service provider. Service provider sends top $\mathcal{K}$ most relevant candidate result set (with reference to the beacon node) back to TP. TP then processes the inputs, performs homomorphic operation, and sends the result back to the congregation. The proposed HYB solution works well for specific queries in which queries are more personalized to the user specific needs.

Location queries can be categorized as generalized or specific queries. A generalized query can also be viewed as a general public query that fulfills the mass requirement, whereas specific query is the one that satisfies individual's need. "Find my nearest retail banking branch of SBI Bank" is the example of specific query, while "Find my nearest bank" is the example of generalized query. In our work it is assumed that user uses the location services to retrieve specific information. The novelty of the proposed hybrid solution is that it exploits the merits of TP and peer group formation without trusting TP as coordinates are kept private by securing them using encryption. Neither query issuer nor TP is aware about the exact locations of the members involved yet it communicates the required results. The rest of the paper is organized as follows: Section 2 highlights the related work. Sections 3 and 4 exhibit the proposed congregation model and homomorphic encryption technique, respectively. The proposed HYB solution is described in Section 5. Section 6 presents performance metrics of HYB solution. Finally, Section 7 concludes the paper.

2. Related Work

A survey of literature in the field of location privacy pertaining to LBS has brought forth several frameworks, architectures, algorithms, and techniques given by numerous researchers and practitioners. Broadly, existing defense mechanisms are based on either of the two architectures: (1)

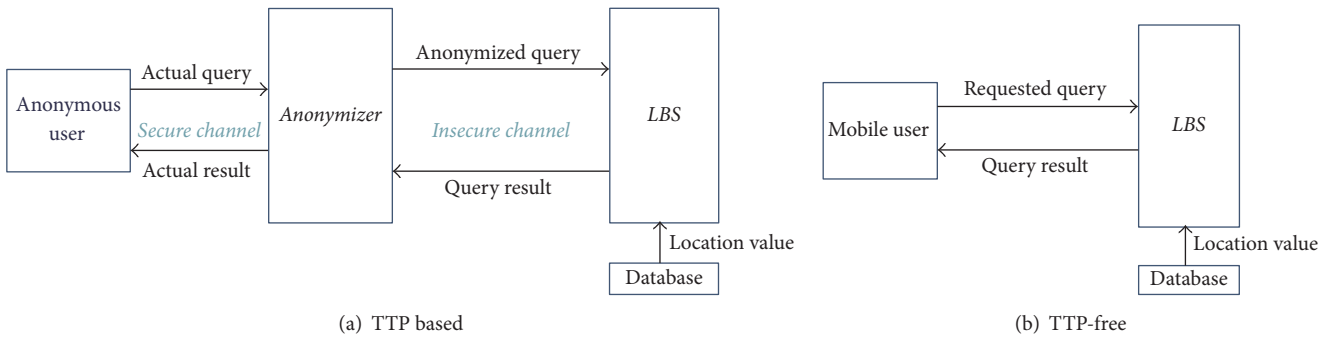


Figure 2: Existing frameworks.

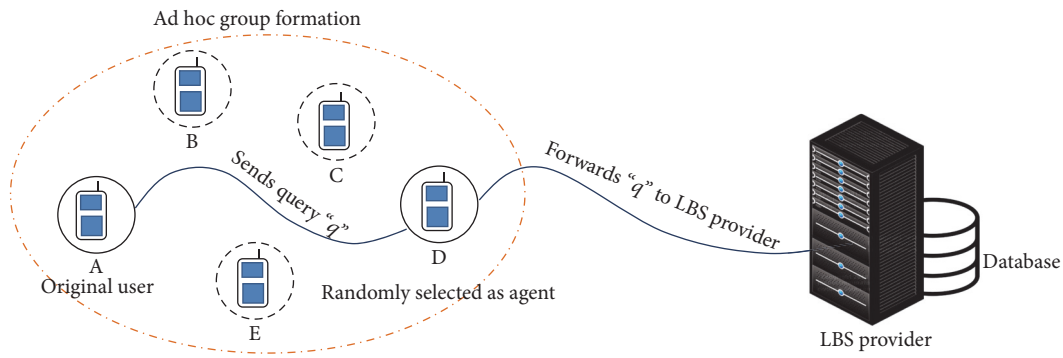


Figure 3: An instance of peer-to-peer spatial cloaking.

centralized architecture or (2) *decentralized architecture*. The setup of these architectures is shown in Figure 2.

In centralized architecture TTP acts as a proxy for service requests and responses between the user and service provider. The greater part of the previous work relies on TTP that mediates user and LBS server [6, 7]. Location anonymity is vastly discussed by [8, 9] in the TTP based architecture. The technique is based on hiding the position data before passing them to the LBS provider. $\mathcal{K}$-anonymity operates by hiding the position of the end user within a set of $\mathcal{K}$ members. Anonymizer includes additional $\mathcal{K} - 1$ users and forwards the anonymized query to LBS provider. It is now difficult for the LBS provider to distinguish the correct user from a set of $\mathcal{K}$ anonymous users. Following are few major constraints due to which TTP based methodologies are losing their ubiquity: (a) The centralized trusted third party can be the system bottleneck, (b) single point of failure is present, (c) a serious privacy threat can occur if the third party is attacked by an adversary, and (d) trusting TP is an absolute vulnerability to the user privacy. Existing cloaking mechanisms are unable to successfully ensure the user's location privacy in a continuous location query scenario (e.g., on the fly route assistance) and can deduce the real location of the client by performing trajectory attacks and dummy continual queries attack [10, 11]. Authors in [12–15] suggest diverse new ideas of using mix zones to mitigate trajectory inference and other attacks. However, it is acceptable but not sufficient to use only technical solutions.

Decentralized architectures, on the other hand, do not consider any intermediate party between users and service provider [16]. The first very basic method proposed to preserve location privacy is through the use of privacy policies [17]. Due the presence of hidden clauses and unsaid policies, this method could not serve the objective of user privacy efficiently for long and as LBS users grew drastically over the years there was a need to have a better and foolproof mechanism. Authors in [18, 19] propose the idea of distributed peer-to-peer communication among mobile users that can freely talk to each other. In this framework, dependence on the third party is eliminated and mobile users are allowed to form an ad hoc network out of which one mobile client is randomly selected as the agent to carry out the communication between querier and LBS server [16]. First, in the query issuer, let user A (refer to Figure 3) glance around and discover the rest of the collaborators to collaborate as a group. The four group members are the mobile users B, C, D, and E; out of them D is randomly chosen as an agent to mediate the communication. Trust among peers plays a profound role in such mechanisms. Evaluation and quantification of trust is another big challenge.

Another TTP-free approach given by [20] proposes a technique to preserve privacy using the concept of geoindistinguishability by adding Laplace noise to the user's Cartesian coordinates. The main objective is to protect issuer's location information while forwarding the aggregate data about the user's area. Differential privacy works on the principle that

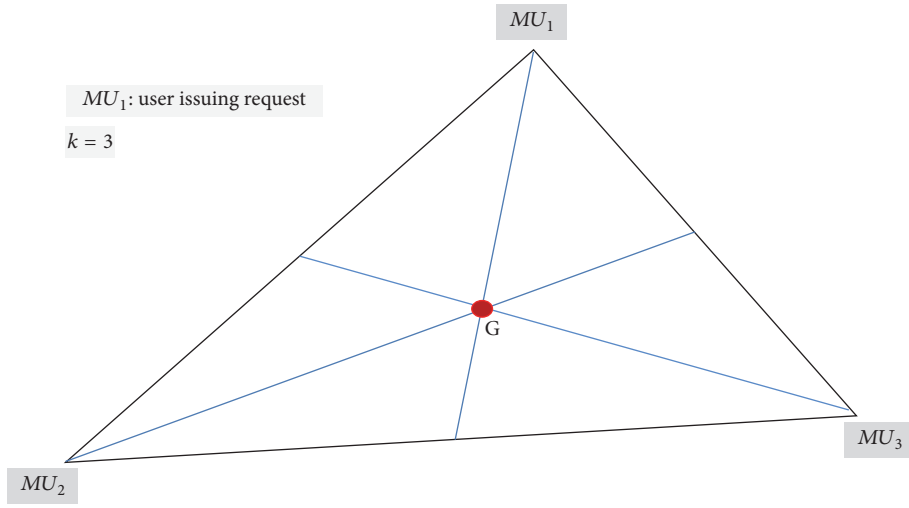


Figure 4: Microaggregation illustration.

modifying one record should have a negligible impact on the outcome of the query. The basic privacy enhancing techniques are first discussed in [21] which protects user's privacy by reducing personal identifiable information without any compromise in system's functionality. Client side obfuscation is also used in which location is repositioned by a random distance and angle of rotation at user's end [22]. The prime shortcoming with such approaches is that different users have different privacy requirement and utility thresholds. Private Information Retrieval (PIR) techniques are also proposed to safeguard the sensitive information like location of the user [23, 24]. These solutions have always been very expensive in terms of operations' computation time, communication cost, and resources needed [25]. Author in [26] first proposed the distributed concept for achieving location privacy in LBS. In this microaggregation based scheme, the major standard of the methodology is to find out the centroid of at least $\mathcal{K}$ perturbed user locations by including zero-mean Gaussian noise and send directly to the LBS database server as shown in Figure 4. The principle issue with [26] is that the centroid of locations with zero-mean Gaussian noise perturbation can be used to deduce the real location if the centroid procedure is repeated several times with the locations of static users. To prevent this problem, authors [27] use a protocol based on privacy homomorphism to ensure that centroid is computed without any knowledge of the real location of the user. Later the similar concept of public key privacy homomorphism is proposed by [28] to achieve location privacy. This is a TTP-free approach in which locations are encrypted under LBS public key and LBS later decrypts them and divides the outcome by the number of users involved to compute centroid. Location decryption by LBS makes this scheme weak and vulnerable to attacks.

The proposed HYB model is dissimilar to these approaches in a way that our solution exploits the merits of both the approaches (TTP based and TTP-free) without disclosing real location of the user anywhere throughout the communication. As of our knowledge the proposed HYB model is the first of its kind that preserves the user's location privacy at two levels, namely, at *proximity* level, while forming congregation, and at *distant* level, while sending encrypted locations to TP and TP performs computation over encrypted input values thereafter.

3. Congregation Model

The model suggests that the query issuer congregates with other $\mathcal{K} - 1$ users as a group and computes the aggregate without knowing the exact locations of the peers. The mobile user mu first broadcasts a *congregate* message to neighboring nodes and shows the intent to use location service. Upon receiving the *congregate* message, willing neighboring nodes send acknowledgment and an ad hoc congregation is formed.

Figure 5 presents the congregation model used in our system model. The mobile user $\mathcal{A}$ considers to be the query issuer node, the one who wants to use location related services. In order to keep the actual location coordinates unknown to others, locations are perturbed by adding a random split to the actual locations. Whole protocol goes as follows.

Protocol 1 (collaborative congregation).

(1) The mobile user mu (the query issuer) adds the random noise to her actual location coordinate (x, y) and generates a tweaked version of the real location, given as

$$\left(x', y'\right) = \left(x + \delta_x, y + \delta_y\right). \quad (1)$$

(2) mu broadcasts a *congregate* message to all neighboring nodes using her tweaked location coordinates to form an ad hoc congregation $\mathcal{C}$.

(3) Willing nodes acknowledge and mu selects $\mathcal{K}$ neighbors to form $\mathcal{C}$. If lesser than $\mathcal{K}$ neighbors acknowledge, step (2) is repeated until required $\mathcal{C}$ is formed

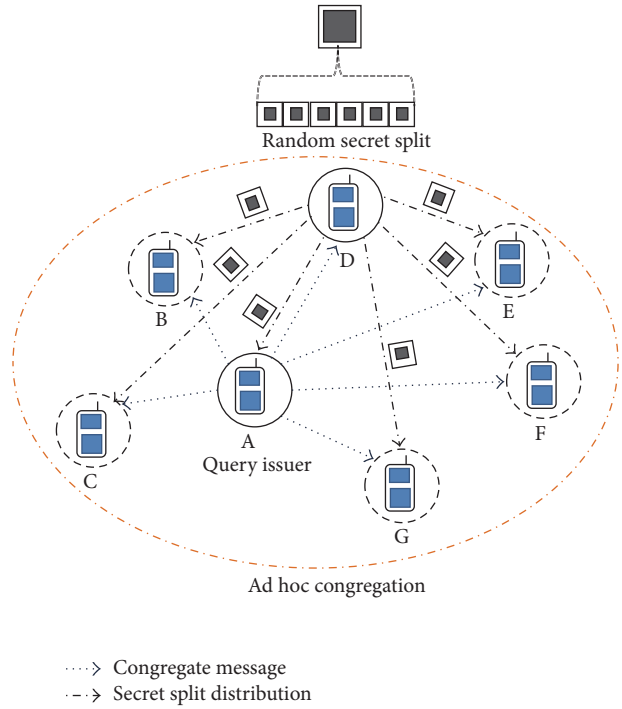


Figure 5: An instance of ad hoc congregation.

which satisfies $\mathscr{K}$. If $\mathscr{K}$ requirement is not fulfilled within a period of Δt, abort and reinitiate the process after $\mathscr{T}$ time interval.

The paucity of enough $\mathscr{K}$ users may introduce unnecessary delay in the query. Therefore, it becomes critical to choose an appropriate value of $\mathscr{K}$. For instance, why would a user feel protected for $\mathscr{K} = 10$ but not the same when $\mathscr{K} = 9$? In many cases $\mathscr{K}$ is demographic dependent, as specifying a larger $\mathscr{K}$ is acceptable for highly populated area, but choosing the same $\mathscr{K}$ value in a deserted area can cause delay in the requested service.

(4) mu randomly selects a node as congregation executor, $E_{\mathbb{C}}$. The responsibility of $E_{\mathbb{C}_i}$ is to facilitate the communication for a congregation $\mathbb{C}_i$.

(5) Now, $E_{\mathbb{C}}$ chooses and splits two sufficiently large random shares $\mathcal{S}_x$ and $\mathcal{S}_y$ such that

$$\begin{aligned} \mathcal{S}_x &= \mathcal{S}_{1,x} + \mathcal{S}_{2,x} + \cdots + \mathcal{S}_{\mathscr{K},x} \\ \mathcal{S}_y &= \mathcal{S}_{1,y} + \mathcal{S}_{2,y} + \cdots + \mathcal{S}_{\mathscr{K},y}. \end{aligned} \tag{2}$$

Splits are generated in such a way that

$$\begin{aligned} \sum_{i\in\mathbb{C}, i=1 \text{ to } \mathscr{K}} \mathcal{S}_{i,x} &= 0, \\ \sum_{i\in\mathbb{C}, i=1 \text{ to } \mathscr{K}} \mathcal{S}_{i,y} &= 0. \end{aligned} \tag{3}$$

(6) $E_{\mathbb{C}}$ sends splits to all the members of $\mathbb{C}$.

(7) Upon receiving the split, each neighbor (including mu) computes a new location (x_p, y_p) by adding the received split value to their actual location coordinates and send them back to $E_{\mathbb{C}}$.

$$\left(x_p, y_p\right) = \left(x_i + \mathcal{S}_{i,x}, y_i + \mathcal{S}_{i,y}\right). \tag{4}$$

(8) $E_{\mathbb{C}}$ computes the centroid of $\mathbb{C}$ defined as

$$\begin{aligned} X_{\mathbb{C}} &= \sum_{i=1 \text{ to } \mathscr{K}} \frac{x_{p,i}}{\mathscr{K}}, \\ Y_{\mathbb{C}} &= \sum_{i=1 \text{ to } \mathscr{K}} \frac{y_{p,i}}{\mathscr{K}}. \end{aligned} \tag{5}$$

(9) $E_{\mathbb{C}}$ passes the centroid $(X_{\mathbb{C}}, Y_{\mathbb{C}})$ to mu and leaves $\mathbb{C}$.

In Figure 5 node $\mathscr{A}$ is the query issuer, while nodes $\mathscr{B}$, $\mathscr{C}$, $\mathscr{D}$, $\mathscr{E}$, $\mathscr{F}$, and $\mathscr{G}$ are the peer members of $\mathbb{C}$. Node $\mathscr{D}$ is randomly selected as $E_{\mathbb{C}}$ and $\mathscr{K} = 6$ is assumed.

Protocol 2 ($\mathbb{C}$ to TP communication).

(a) mu encrypts $(X_{\mathbb{C}}, Y_{\mathbb{C}})$ by her own public key (pk) and gets the encrypted value $\mathscr{E}(X_{\mathbb{C}}, Y_{\mathbb{C}})$.

(b) mu generates the query describes as

$$\mathcal{Q}\text{: } \left\langle \mathscr{E}\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right), \mathscr{BS}_{\mathbb{C}}, \mathscr{K}, \text{"specific search string"} \right\rangle, \tag{6}$$

where $\mathscr{BS}_{\mathbb{C}}$ is the identifier of the base station under which umbrella $\mathbb{C}$ is formed and $\mathscr{K}$ is the anonymity parameter specified by mu.

4. Homomorphic Encryption

An efficient and straightforward remedy to preserve user privacy in location (or any cloud based) services is to encrypt the information before sending to the service provider. Nonetheless, this straightforward arrangement has a critical downside in that if the information is scrambled utilizing a routine encryption method, the service provider (or cloud) can not process the information without decrypting it first. Obviously, sharing the secret decryption key with service provider again puts the same problem of privacy at stake.

In order to eliminate the mentioned problem of user privacy, a homomorphic encryption technique is used that permits some calculation to be performed specifically on encrypted information without any decryption [29].

Broadly, homomorphic encryption can be defined as follows: Suppose $\mathscr{P}$ represents the plain texts set, $\mathscr{C}$ represents corresponding set of cipher texts, and $\mathscr{ENC}$ denotes given encryption function; the cryptosystem is said to be *homomorphic* if it satisfies

$$\mathscr{ENC}\left(p_1 \odot_{\mathscr{P}} p_2\right) \longleftarrow \mathscr{ENC}\left(p_1\right) \odot_{\mathscr{C}} \mathscr{ENC}\left(p_2\right), \quad \forall p_1, p_2 \in \mathscr{P}, \tag{7}$$

where $\odot_{\mathscr{P}}$ in $\mathscr{P}$ and $\odot_{\mathscr{C}}$ in $\mathscr{C}$ are some operators. We call such disposition an *additive homomorphism* if we use addition operators and a *multiplicative homomorphism* if we use multiplication operators.

Homomorphism supports both types of encryption scheme: a symmetric key encryption and an asymmetric key encryption. There are three key elements required to specify a public key (or asymmetric) cryptosystem: an encryption algorithm $\mathscr{ENC}_{\mathrm{pk}}$, a decryption algorithm $\mathscr{DEC}_{\mathrm{sk}}$, and a key-pair generator algorithm that produces the public key and secret key (or private key) pair. The $\mathscr{ENC}_{\mathrm{pk}}$ algorithm takes the plain text and produces the encrypted text using public key pk. The output of $\mathscr{ENC}_{\mathrm{pk}}$ becomes input for $\mathscr{DEC}_{\mathrm{sk}}$ algorithm and encrypted text decrypts using the secret key sk. Homomorphic encryption permits calculations to be done on encrypted data (or cipher text). The computations are done in such a way that result when decrypted (using sk) matches the results of operations performed on the plain text.

Our proposed hybrid model takes the advantage of the homomorphic encryption property which allows the operations to be performed over encrypted data without decrypting it. Unlike existing addition and multiplication operations over encrypted data, we suggest difference (or subtraction) operation over encrypted data. However, existing cryptosystem that supports additive homomorphism [30, 31] is used to perform the proposed operation.

5. Proposed Hybrid Model

Hybrid model is built upon the concept of collaborative congregation and use of third party to mediate the results in a more effective way. The hybrid scheme appears to be centralized (due to TP) yet decentralized as no user locations are disclosed even to TP during entire communication. TP is used to provide computational support that makes the overall communication faster and efficient.

Following are the phases of our proposed scheme.

Phase 1 (ad hoc congregation $\mathbb{C}$). Mobile user mu, who wants to avail the location service, first broadcasts a *congregate* message to neighbors until required $\mathscr{K}$ users respond. This phase ends with a formation of $\mathbb{C}$ and a computed pair of $(X_{\mathbb{C}}, Y_{\mathbb{C}})$ at mu as per Protocol 1 of Section 3. mu encrypts the centroid coordinates $(X_{\mathbb{C}}, Y_{\mathbb{C}})$ with her own public key (pk) and forwards the query $\mathscr{Q}$ to TP as per Protocol 2 of Section 3.

Phase 2 (communication from TP to LBS and back). Once TP receives $\mathscr{Q}$, it strips off $\mathscr{E}(X_{\mathbb{C}}, Y_{\mathbb{C}})$ and forwards remaining $\mathscr{Q}$ to LBS provider. According to $BS_{\mathbb{C}}$ relevance, LBS look into the assisted database and returns top $\mathscr{K}$ candidate results to the TP given as

$$\mathrm{CR}: \left\langle \left(x_1, y_1\right), \left(x_2, y_2\right), \ldots, \left(x_k, y_k\right) \right\rangle, \tag{8}$$

where CR represents the candidate result.

Phase 3 (TP computation). TP preprocesses the data by multiplying all the items of candidate result set by a constant (-1) and encrypts this modified CR by mu's public key.

$$\mathscr{E}(\mathrm{CR}): \left\langle \mathscr{E}_{\mathrm{pk}}\left(x_1, y_1\right), \mathscr{E}_{\mathrm{pk}}\left(x_2, y_2\right), \ldots, \mathscr{E}_{\mathrm{pk}}\left(x_k, y_k\right) \right\rangle. \tag{9}$$

TP now has encrypted centroid coordinates $\mathscr{E}(X_{\mathbb{C}}, Y_{\mathbb{C}})$, and encrypted set of candidate results $\mathscr{E}(\mathrm{CR})$. The motive is to find the distance between the target point (centroid here) and the relevant points sent by the LBS provider so that the proximity of two can be measured. An additive homomorphic encryption is then applied to $(X_{\mathbb{C}}, Y_{\mathbb{C}})$ and each item of encrypted candidate result set separately given as

$$\begin{aligned} \mathscr{E}\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) \cdot \mathscr{E}_{\mathrm{pk}}\left(x_1, y_1\right) &= \mathscr{E}\left(\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) + \left(x_1, y_1\right)\right) \\ \mathscr{E}\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) \cdot \mathscr{E}_{\mathrm{pk}}\left(x_2, y_2\right) &= \mathscr{E}\left(\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) + \left(x_2, y_2\right)\right) \\ &\vdots \\ \mathscr{E}\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) \cdot \mathscr{E}_{\mathrm{pk}}\left(x_k, y_k\right) &= \mathscr{E}\left(\left(X_{\mathbb{C}}, Y_{\mathbb{C}}\right) + \left(x_k, y_k\right)\right). \end{aligned} \tag{10}$$

TP forwards the encrypted results and CR (in plain text) to mu. The purpose of having TP between mu and LBS is to perform certain computation such that the information retrieval becomes faster and relevant that too without losing any location privacy.

Phase 4 (decryption at mu). The mu has $\mathscr{K}$ encrypted values that can be viewed as the distances between the encrypted coordinates sent by $\mathbb{C}$ and the candidate result points sent by the LBS provider. mu deciphers them using her own secret key (sk). Let decryption gives the set of distances $\mathscr{D}$. Clearly, the minimum, $\min(\mathscr{D})$, among all distance values is the most relevant result. mu keeps the corresponding location

(1) Function: Communication using Hybrid System Model
(2) //*Phase 1: Ad hoc Congregation* $\mathbb{C}$
(3) *Let mobile user "mu" starts the query and* $\mathcal{K}$ *represents the number of users required to form* $\mathbb{C}$
(4) *Let* $\mathcal{S}$ *be the set to count numbers of neighbors responded*
(5) *Initially,* $\mathbb{C}(\mathcal{K}) = \emptyset$, $\mathcal{S} = 0$, $i = 0$
(6) *Let mu's actual location coordinates* $= (x, y)$
(7) $(x', y') = (x + \delta_x, y + \delta_y)$
(8) **while** $(i \leq \mathcal{K})$ **do**
(9) *mu broadcasts a CONGREGATE message to neighbors*
(10) *Let* $\mathcal{S}$ *users acknowledge mu*
(11) $i = |\mathcal{S}|$
(12) $\mathbb{C}(\mathcal{K}) = \mathbb{C}(\mathcal{K}) \cup$ *nodes in S*
(13) $i = i + |\mathcal{S}|$
(14) **return** $\mathbb{C}(\mathcal{K})$ //congregation formed
(15) **end**
(16) mu chooses a random node as congregation executor $E_{\mathbb{C}}$, $E_{\mathbb{C}} \in \mathbb{C}(\mathcal{K})$
(17) CALL Secret_Split_Function;
(18) *Let set* $X_p(\mathcal{K})$ *and* $Y_p(\mathcal{K})$ *holds the perturbed locations received after secret splitting*
(19) CALL Centroid_Function;
(20) $E_{\mathbb{C}}$ forwards $X_{\mathbb{C}}, Y_{\mathbb{C}}$ to mu and leaves $\mathbb{C}$;
(21) mu generates (pk, sk) pair
(22) $\mathscr{E}_{\text{pk}}(X_{\mathbb{C}}, Y_{\mathbb{C}})$ //encrypted points
(23) //*Phase 3: Computation performed at TP*
(24) (X_L, Y_L) = CALL TP_Computation-I;
(25) $\mathscr{E}_{\text{pk}}(X_L, Y_L)$ //Encryption using mu's pk
(26) CALL TP_Computation-II;
(27) //Phase 4: Decryption at mu
(28) $\mathscr{D}_{\text{sk}}((X_{\mathbb{C}} + X_L), (Y_{\mathbb{C}} + Y_L))$ //Decryption using mu's sk
(29) *Let* X_D, Y_D *be the set of distance difference received on decryption*
(30) CALL Min_dist (X_D, Y_D)
(31) *Broadcast Results to all members of* $\mathbb{C}$

Algorithm 1: HYB solution.

coordinate against min($\mathscr{D}$) and sends remaining results to all the members of $\mathbb{C}$.

Considerations and Assumptions

(a) The utilized mobile devices are Location Based Services enabled and have the ability to determine their approximate location.

(b) The TP possess required computation power and processing potential.

(c) Location queries are sporadic, pull-based, and specific in nature.

(d) Generation of ⟨Public-Private⟩ key pair at mu is implicit.

Algorithm Description. The algorithm, HYB solution, gives pseudocode for the overall communication of our proposed hybrid system model. A congregation is formed (lines (7)–(15) in Algorithm 1), a pair of coordinates are computed (lines (16)–(19) in Algorithm 1), and the encryption is performed (lines (21)–(23) in Algorithm 1) over computed coordinates during Phase 1 of HYB solution. Phase 2 fetches the candidate result from LBS to TP. In Phase 3, candidate result is first modified (line (24) in Algorithm 1) and then encrypted (line (25) in Algorithm 1) before applying homomorphic operation (line (26) in Algorithm 1) over encrypted inputs. Decryption is performed in Phase 4 (line (28) in Algorithm 1) and the minimum is calculated (line (30) in Algorithm 1) to get optimum result. Algorithms 2, 3, 4, 5, and 6 give the pseudocodes for the suboperations: splitting the random secret, centroid computation, input preprocessing, homomorphic encryption, and finding minimum value from the result set, respectively.

6. Empirical Evaluation

We develop the simulation scenario and implemented the same in Java. We run it on an Intel Core 3.20 GHz machine with 4 GB of RAM running Linux OS. We experimented the

(1) Function: Secret Splitting Sharing
(2) $E_{\mathbb{C}}$ *chooses and split sufficiently large two random shares* $\mathcal{S}_x$ *and* $\mathcal{S}_y$ *s.t.*
$\sum_{i\in\mathbb{C}, i=1 \text{ to } \mathcal{K}} \mathcal{S}_{i,x} = 0, \quad \sum_{i\in\mathbb{C}, i=1 \text{ to } \mathcal{K}} \mathcal{S}_{i,y} = 0$
(3) $E_{\mathbb{C}}$ *sends separate split values to every node* $\in \mathbb{C}(\mathcal{K})$
(4) **foreach** node $\in \mathbb{C}(\mathcal{K})$ **do**
(5) **for** $i = 1; i_{\mathbb{i}}=\mathcal{K}; i++$ **do**
(6) $(x_p, y_p) = (x_i + \mathcal{S}_{i,x}, y_i + \mathcal{S}_{i,y})$
(7) **end**
(8) **return** (x_p, y_p)
(9) **end**

Algorithm 2: Secret_Split_Function.

(1) Function: Centroid Computation
(2) **foreach** $x \in X_p(\mathcal{K})$ *and* $y \in Y_p(\mathcal{K})$ **do**
(3) $i = 1, Temp_x = Temp_y = 0$
(4) **while** $i <= \mathcal{K}$ **do**
(5) $Temp_x = Temp_x + x_i;$
(6) $Temp_y = Temp_y + y_i;$
(7) $i++;$
(8) **end**
(9) $X_{\mathbb{C}} = \frac{Temp_x}{\mathcal{K}}, Y_{\mathbb{C}} = \frac{Temp_y}{\mathcal{K}}$
(10) **return** $(X_{\mathbb{C}}, Y_{\mathbb{C}})$
(11) **end**

Algorithm 3: Centroid_Function.

(1) Function: Coordinate Pre-processing
(2) *Let set* $X_l(\mathcal{K})$ *and* $Y_l(\mathcal{K})$ *be the points provided by LBS*
(3) *Let set* X_l' *and* Y_l' *be the points modified by TP*
(4) *Initially*, $i = 0, X_l' = Y_l' = 0$
(5) **foreach** $x \in X_l(\mathcal{K})$ *and* $y \in Y_l(\mathcal{K})$ **do**
(6) **while** $i \leq \mathcal{K}$ **do**
(7) $x_i' = (-1) * x_i, y_i' = (-1) * y_i;$
(8) $i++;$
(9) **end**
(10) $X_l' = X_l' \cup x_i', Y_l' = Y_l' \cup y_i'$
(11) **end**
(12) return (X_l', Y_l')

Algorithm 4: TP_Computation-I.

(1) Function: Computing point differnce
(2) *input*: $X_{\mathbb{C}}, Y_{\mathbb{C}}$ *and* X_L, Y_L
(3) Apply *Paillier* Homomorphic Encryption
(4) **return** $((X_{\mathbb{C}} + X_L), (Y_{\mathbb{C}} + Y_L))$

Algorithm 5: TP_Computation-II.

(1) Function: Finding location with minimum distance
(2) *Let* MIN *represents the minimum element of the list,* $i = 2$ **foreach** *element of* X_D, Y_D **do**
(3) (X_1, Y_1) = MIN **while** $i <= \mathcal{K}$ **do**
(4) **if** (X_i, Y_i) < MIN **then**
(5) MIN = (X_i, Y_i)
(6) $i++;$
(7) **end**
(8) **else**
(9) $i++;$
(10) **end**
(11) **end**
(12) **return** MIN;
(13) **end**

Algorithm 6: Min_Dist.

Table 1: Parameters used with description.

Parameter	Description	Values used
$\mathcal{K}$	Anonymity parameter	1, 5, 10, 20, 40, 100, 150, 200
$\mathcal{N}$	Key size (in bits)	512, 1024, 2048, 4096
Review period	Time interval between two consecutive runs of the algorithm	90 s
Total run count	Number of times the algorithm runs for a particular combination of parameters used	100
Input size	Size of an input item	4 KB

performance with different variations in anonymity parameter and key size. Performance metrics is measured in average computation time taken by the processes.

6.1. Parameters Description. Results are evaluated for different values of parameters. Table 1 highlights the brief description of the parameters used.

6.2. Anonymity Parameter and Key Size Impact over TP Computation-II. The first experiment explores the impact of anonymity parameter with different key sizes over the performance of the system in terms of the computation time. The algorithm TP Computation-II computes the homomorphic encryption.

Analysis. Figure 6 shows the average time taken by TP to perform operations over encrypted data. It can be seen that time taken is very less (less than a second) for those combinations where key size ($\mathcal{N}$) and $\mathcal{K}$ are low. As we move left to right through x-axis in the graph, the time increases beyond acceptable threshold and makes the framework costly in terms of time for higher values of $\mathcal{N}$ and $\mathcal{K}$.

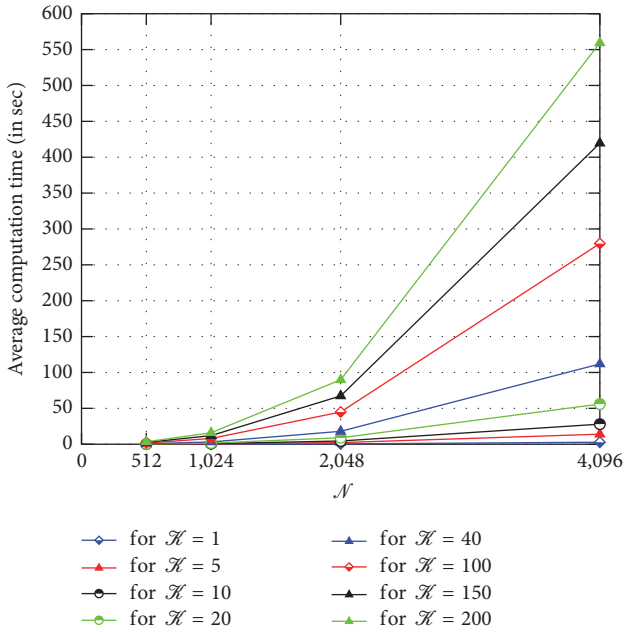


Figure 6: Anonymity parameter and key size impact over TP Computation-I.

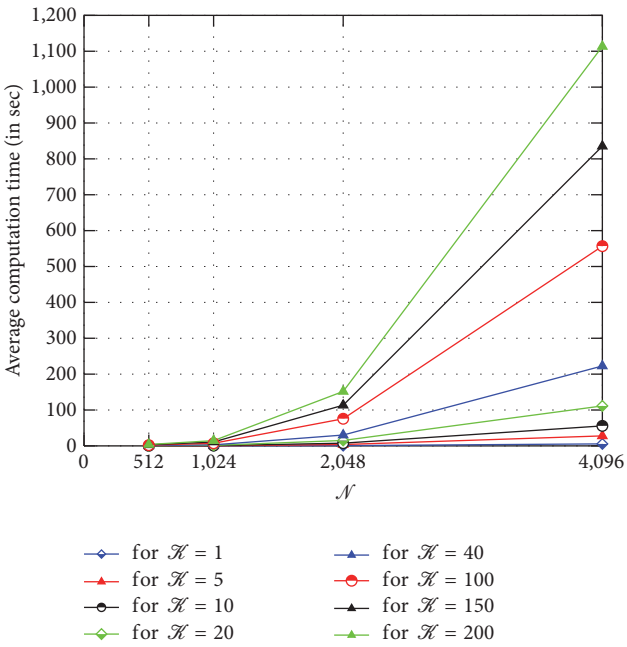


Figure 7: Anonymity parameter and key size impact over decryption.

6.3. Anonymity Parameter and Key Size Impact over Decryption Computation at mu. This evaluation shows the time taken to decrypt the encrypted results. Decryption is performed using mu's secret key which is secure and not shared with any other party.

Analysis. Figure 7 shows the average computation time for decryption. The effect of $\mathcal{K}$ and $\mathcal{N}$ is more or less similar as in the case discussed before. It is clear that computation time is lesser for smaller $\mathcal{K}$ and $\mathcal{N}$ values; on the other hand, computation cost becomes exorbitantly expensive for higher $\mathcal{K}$, $\mathcal{N}$ values combination.

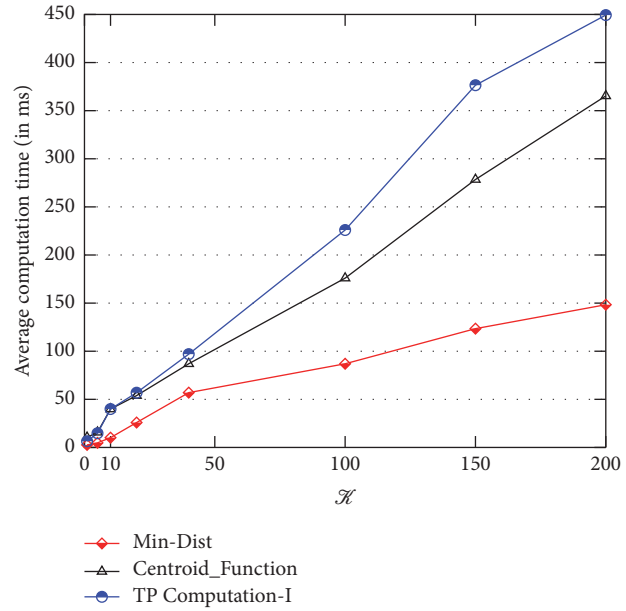


Figure 8: Miscellaneous computation time dependence over anonymity parameter $\mathcal{K}$.

6.4. Effect of Size of ζ over Miscellaneous Computation. Min-Dist is used to calculate the minimum among all the values received after decryption. TP Computation-I preprocesses the input and Centroid Function computes the centroid of locations. These processes also contribute to the overall time of HYB solution.

Analysis. Figure 8 shows that, for lower $\mathcal{K}$ values, the computation time is lower. However, time taken for higher $\mathcal{K}$ (150 and 200) is much lesser compared to the time taken by TP Computation-II and becomes less significant when added to the overall computation cost.

The value of $\mathcal{K}$ specified by the mobile user mu and the key size used for encryption impacts the overall computation time to a large extent. The balanced combination of these two parameters produces the optimum results. Moreover, the public key encryption enabled the secure communication as no key distribution is now needed. As the location data is encrypted under mu's public key and decryption takes place at mu with the secret key she has, it makes the overall solution secure and reliable.

7. Conclusion

This paper first addressed the issues in TTP based and TTP-free frameworks and presented a hybrid solution that makes effective use of the advantages both the approaches possess, to preserve location privacy of the user through congregation and homomorphic encryption. The novelty of

the proposed HYB solution lies in the fact that involvement of third party is introduced to perform computations only and TP has no knowledge of the user's real location. A congregation scheme is also suggested that helps the mobile user to compute centroid of all the users involved, that too without knowing anyone's actual location. Homomorphic encryption technique is used with a modified input data in order to take most out of it. We have analyzed the performance of our model for various key sizes and for different values of anonymity parameter. Our scheme works well when key size and anonymity parameter are in a certain range. The proposed HYB model preserves the user's location privacy at two levels, namely, at *proximity* level, while forming congregation, and at *distant* level, while sending encrypted location to TP.

Conflicts of Interest

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

References

[1] D. Wells, N. Beck, A. Kleusberg et al., *Guide to GPS Positioning*, Canadian GPS Associates, New Brunswick, Canada, 1987.

[2] A. Jafarnia-Jahromi, A. Broumandan, J. Nielsen, and G. Lachapelle, "GPS vulnerability to spoofing threats and a review of antispoofing techniques," *International Journal of Navigation and Observation*, vol. 2012, Article ID 127072, 2012.

[3] M. F. Mokbel, "Privacy in location-based services: state-of-the-art and research directions," in *Proceedings of the 8th International Conference on Mobile Data Management (MDM '07)*, p. 228, IEEE, Mannheim, Germany, May 2007.

[4] J. Krumm, "Inference attacks on location tracks," in *Pervasive Computing*, pp. 127–143, Springer, Berlin, Germany, 2007.

[5] D. Song and K. Park, "A privacy-preserving location-based system for continuous spatial queries," *Mobile Information Systems*, vol. 2016, Article ID 6182769, 9 pages, 2016.

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

[7] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *Proceedings of the 32nd International Conference on Very Large Data Bases*, pp. 763–774, VLDB Endowment, 2006.

[8] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, pp. 31–42, ACM, May 2003.

[9] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Mobihide: a mobilea peerto-peer system for anonymous location-based queries," in *Advances in Spatial and Temporal Databases*, pp. 221–238, Springer, Berlin, Germany, 2007.

[10] E. K. Wang and Y. Ye, "A new privacy-preserving scheme for continuous query in location-based social networking services," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 979201, 2014.

[11] M. Zhou, X. Li, and L. Liao, "On preventing location attacks for urban vehicular networks," *Mobile Information Systems*, vol. 2016, Article ID 5850670, 13 pages, 2016.

[12] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran, "AMOEBA: robust location privacy scheme for VANET," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1569–1589, 2007.

[13] J. Freudiger, M. Raya, M. Félegyházi, P. Papadimitratos, and J.-P. Hubaux, "Mix-zones for location privacy in vehicular networks," in *Proceedings of the ACM Workshop on Wireless Networking for Intelligent Transportation Systems (WiN-ITS '07)*, Vancouver, Canada, 2007.

[14] F. Kargl and J. Petit, "Security and privacy in vehicular networks," in *Vehicular Communications and Networks: Architectures, Protocols, Operation and Deployment*, pp. 171–189, 2015.

[15] Y. Gai, J. Lin, and B. Krishnamachari, "Security and privacy in vehicular networks," *Cognitive Vehicular Networks*, pp. 151–166, 2016.

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

[17] M. Langheinrich, "Privacy by designprinciples of privacy-aware ubiquitous systems," in *Ubicomp 2001: Ubiquitous Computing*, pp. 273–291, Springer, Berlin, Germany, 2001.

[18] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Prive: anonymous locationbased queries in distributed mobile systems," in *Proceedings of the 16th International Conference on World Wide Web*, pp. 371–380, ACM, 2007.

[19] H. Zhangwei and X. Mingjun, "A distributed spatial cloaking protocol for location privacy," in *Proceedings of the 2nd International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC '10)*, vol. 2, pp. 468–471, April 2010.

[20] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: differential privacy for location-based systems," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 901–914, ACM, Berlin, Germany, November 2013.

[21] R. Koorn, H. van Gils, J. ter Hart, P. Overbeek, R. Tellegen, and J. Borking, *Privacy Enhancing Technologies, White Paper for Decision Makers*, 2004.

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

[23] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: anonymizers are not necessary," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*, pp. 121–132, ACM, June 2008.

[24] A. Khoshgozaran, H. Shirani-Mehr, and C. Shahabi, "SPIRAL: a scalable private information retrieval approach to location privacy," in *Proceedings of the 9th International Conference on Mobile Data Management Workshops (MDMW '08)*, pp. 55–62, April 2008.

[25] A. Khoshgozaran and C. Shahabi, "Private information retrieval techniques for enabling location privacy in location-based services," in *Privacy in Location-Based Applications*, pp. 59–83, Springer, 2009.

[26] J. Domingo-Ferrer, "Microaggregation for database and location privacy," in *Next Generation Information Technologies and Systems*, pp. 106–116, Springer, 2006.

[27] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 308–318, Springer, 1998.

[28] A. Solanas and A. Martínez-Ballesté, "A TTP-free protocol for location privacy in location-based services," *Computer Communications*, vol. 31, no. 6, pp. 1181–1191, 2008.

[29] R. Rothblum, "Homomorphic encryption: from private-key to publickey," in *Proceedings of the Theory of Cryptography Conference*, pp. 219–234, Springer, Providence, RI, USA, March 2011.

[30] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '99)*, pp. 223–238, Springer, Prague, Czech Republic, 1999.

[31] I. Damgård and M. Jurik, "A generalisation, a simplification and some applications of Paillier's probabilistic public-key system," in *International Workshop on Public Key Cryptography*, pp. 119–136, Springer, 2001.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 6195024, 10 pages
https://doi.org/10.1155/2017/6195024

Research Article

Network Access Control for Location-Based Mobile Services in Heterogeneous Wireless Networks

Dae-Young Kim,[1] Dae-sik Ko,[2] and Seokhoon Kim[3]

[1]*Department of Software Engineering, Changshin University, Changwon, Republic of Korea*
[2]*Department of Electronic Engineering, Mokwon University, Daejeon, Republic of Korea*
[3]*Department of Computer Software Engineering, Soonchunhyang University, Asan, Republic of Korea*

Correspondence should be addressed to Seokhoon Kim; seokhoon@sch.ac.kr

Received 28 November 2016; Accepted 4 April 2017; Published 23 April 2017

Academic Editor: Subramaniam Ganesan

Recent advances in information communication technology and software have enabled mobile terminals to employ various capabilities as a smartphone. They adopt multiple interfaces for wireless communication and run as a portable computer. Mobile services are also transferred from voice to data. Mobile terminals can access Internet for data services anytime anywhere. By using location-based information, improved mobile services are enabled in heterogeneous networks. In the mobile service environment, it is required that mobile terminals should efficiently use wireless network resources. In addition, because video stream becomes a major service among the data services of mobile terminals in heterogeneous networks, the necessity of the efficient network access control for heterogeneous wireless networks is raised as an important topic. That is, quality of services of the location-based video stream is determined by the network access control. Therefore, this paper proposes a novel network access control in the heterogeneous wireless networks. The proposed method estimates the network status with Naïve Bayesian Classifier and performs network access control according to the estimated network status. Thus, it improves data transmission efficiency to satisfy the quality of services. The efficiency of the proposed method is validated through the extensive computer simulation.

1. Introduction

Nowadays, mobile terminals adopt multiple network interfaces such as cellular, WiFi, and Bluetooth. They are widely used as smartphones and enable various services in heterogeneous networks. Services for the mobile terminals are changing to data-based services from circuit-based services. In the data-based services, there exist web services, online game, video streaming, and so on. Among them, demand for high quality video streaming is increased. Video traffic will be occupied by 75% of total mobile data traffic in 2020 [1]. Particularly, the growth of location-based video services is expected. The demand of this mobile service user causes mobile terminals to find better network connection for better services. Operating systems for the mobile terminals such as android of google and iOS of apple support controlling multiple network interfaces. Application services provide various services using the multiple network interfaces. Thus, users can use data services that they want, through various network interfaces of mobile terminals, anytime anywhere.

Over the Top (OTT) services are mobile TVs which provide video contents over mobile Internet. Using location-based information of mobile terminals, video clip for advertisement can be provided to service users. Through the location-based video service, marketing effects can be maximized. In this mobile service environment, because 4G LTE networks are widely spread and public WiFi networks are increased, communication environments for mobile Internet are improved. However, data traffic of the high quality videos grows very fast and radio resources to serve the data traffic are gradually lacking. Thus, OTT service providers are interested in exploiting multiple networks to connect Internet. The OTT services mainly employ cellular and WiFi network as shown in Figure 1. For their quality of services, OTT service providers take network access control between cellular and WiFi network into account. Several researches for

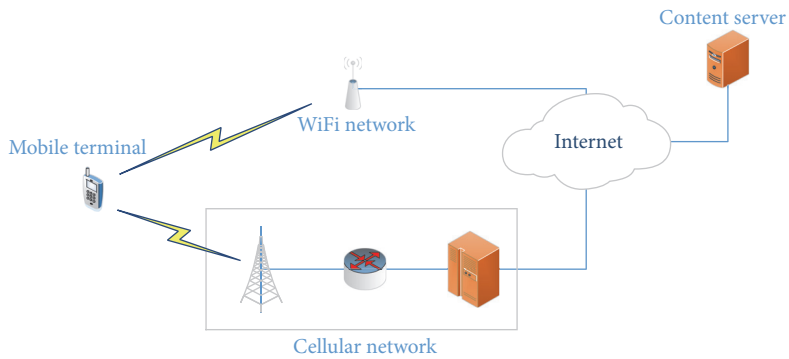


Figure 1: Network system architecture for the OTT services.

the OTT services approached efficient usage of the cellular and WiFi network to improve the quality of services and user satisfaction.

In OTT services, 98% of video traffic uses Hypertext Transfer Protocol (HTTP) to transmit data traffic [2, 3]. It is possible that a mobile terminal requests video traffic in byte-range by HTTP Range Requests [4–7]. Media content servers transmit video traffic in the requested byte-range. Through the HTTP Range Requests, requesting data in byte-range according to the network interface is enabled. That is, mobile terminal chooses a target network and can receive video traffic in the requested bytes by a media player. Because the HTTP Range Requests technology deals with data traffic as a byte block (i.e., chunk), segmentation and assembly of data traffic are available. Thus, it can be used to change network interfaces during video streaming and to adjust traffic load of network interfaces in a mobile terminal.

Multimedia applications of mobile services employ multiple wireless networks and thereby selecting the network interface in a mobile terminal according to the wireless channel status is crucially affected on the quality of services. In case of cellular network, it provides connectivity in wide area; however, it shows fluctuation in data rate. WiFi network has the weak point in aspect of providing seamless connectivity. Therefore, efficient network control between cellular and WiFi [8] is required, and providing services by recognizing the network status and efficient controlling network interfaces is raised as an important challenge.

According to user policies, network interfaces of mobile terminals can be diversely exploited. If the goal of users is cost effective service, WiFi network can be preferentially selected to provide video streaming. Relatively, if the goal is seamless connectivity, cellular network can be selected by priority [9–11]. In this paper, however, the network interface for a streaming service is determined according to the quality of services (i.e., content rate of a media player in a mobile terminal) instead of the characteristics of wireless networks. Moreover, not only is the network switched between cellular and WiFi network, but also the simultaneous usage of cellular and WiFi network is solved. The Naïve Bayesian Classifier, which is based on statistics of successful transmission rate and signal strength in a mobile terminal, is exploited to be aware of channel status of wireless networks. The method to select the proper network interface according to the channel status is proposed in order to satisfy the quality of services of a media player. By learning using the Naïve Bayesian Classifier, the proposed method is expected to improve the estimation accuracy of the network status and efficient decision for network interface selection can be carried out from that.

The remainder of paper is organized as follows. Section 2 discusses the related work on network selection method specific to heterogeneous wireless networks. The proposed network access control method for heterogeneous wireless networks is presented in Section 3. Section 4 presents the performance evaluation. Finally, Section 5 concludes the paper.

2. Related Work

OTT services such as mobile TV become widely spread and location-based services are added to the OTT services in order to provide improved services and to maximize business effect of service providers. They are served by heterogeneous wireless networks which consist of cellular and WiFi. The heterogeneous wireless networks focus on the efficient network selection to provide the best services for mobile terminals. There are two types of network selection methods (i.e., terminal-side selection and network-side selection). In the terminal-side network selection, a terminal classifies several elements such as mobility, traffic, and cost. Then, it assigns scores to the elements. According to the characteristics of given services, the terminal combines the scores and applies the combined score to the network selection [12]. In the network-side selection, Common Radio Resource Management (CRRM) module, which is a network device, manages radio resources of whole wireless networks. It selects a network for mobile terminals and assigns proper radio resources to the mobile terminals [13–16]. In the network selection, if a network is selected only according to elements of user preference, the change of network status cannot be adaptively reflected. Thus, the elements of network status should be considered to choose a network for a given service.

There are several policies to select the proper networks: the network selection policy to maximize performance of mobile terminals, the network selection policy to minimize usage of the cellular traffic, and the network selection policy

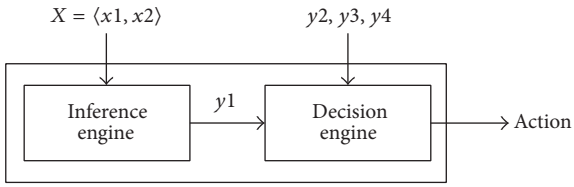


Figure 2: Architecture of the proposed network access control.

to conserve energy consumption of mobile terminals [9]. In the first policy, a mobile terminal monitors data rate of wireless networks and compares the data rates per the given time. Then the mobile terminal selects a network with bigger data rate. In the second policy, if WiFi network is available, a mobile terminal chooses the WiFi. However, when the signal strength of the WiFi is less than a certain threshold, the mobile terminal changes the WiFi network to cellular network. In the last policy, a mobile terminal uses cellular network at first. Then if traffic usage is larger than a certain threshold, it changes the cellular network to WiFi network. In general, a mobile terminal in the cellular network consumes more energy to exchange data traffic than the WiFi network. Thus, to reduce consumed energy in the mobile terminal, the amount of traffic usage in the cellular network should be restricted.

When the channel status of wireless networks is frequently changed, data rates of the wireless networks are the crucial factor for the efficient data transmission. Mobile terminals should satisfy the required data rates of services. In [8], the method combining the bandwidth of both cellular (LTE) network and WiFi network for services was proposed.

Existing methods in heterogeneous wireless networks to select a network in a mobile terminal considered user preference, amount of data usage, data rate, and so on. However, these methods cannot reflect both buffer status of media players and the network status. In addition, they cannot provide stable data reception and cause service problems such as media pause. Thus, in this paper, in order to be aware of the network status, Naïve Bayesian Classifier, which is based on statistics of data rate, is exploited. Through the classifier, the network status can be estimated. Then, according to the estimated network status and buffer status for a media player, the mobile terminal selects the proper network. In addition, the mobile terminal can use both cellular and WiFi networks simultaneously. In proposed method, the mobile terminal can provide the proper data transmission for its services.

3. The Proposed Radio Access Control

The proposed method consists of the inference engine and the decision engine. The inference engine estimates network status using Naïve Bayesian Classifier. The decision engine decides behaviors for the network access control. The proposed method is periodically operated for the network access control. Figure 2 represents the architecture of the proposed network access control.

The inference engine manages statistics data for successful transmission rate (x_1) and signal strength (x_2) in WiFi networks. It estimates the network status of WiFi networks through the Bayesian Inference with the statistics data. The estimation of WiFi status is for the network status that services are available. Then, according to the result of the estimation (y_1), the decision engine performs the network access control with buffer status of a media player (y_2), stability of the WiFi network (y_3), and system status for incoming/outgoing traffic (y_4).

3.1. WiFi Status Estimation in the Inference Engine. The inference engine employs Naïve Bayesian Classifier. The Naïve Bayesian Classifier is based on the Bayes rule and it is a widely used supervised learning algorithm (supervised learning is widely applied to wireless networks to estimate the variance of wireless resources and network environment [17–23]). It is used to estimate the most possible state from probability by the a priori statistic information. Therefore, the more the training data by experience, the better the decision accuracy. The probability of the most possible state can be obtained by the Bayes rule. The Naïve Bayesian Classifier calculates the posterior probability of each state and chooses the state with the largest probability. It is represented as

$$\begin{aligned} v &= \arg\max_{Y} P(Y \mid X) = \arg\max_{Y} \frac{P(X \mid Y)\, P(Y)}{P(X)} \\ &= \arg\max_{Y} P(X \mid Y)\, P(Y). \end{aligned} \tag{1}$$

In the Naïve Bayesian Classifier, when attributes X are given, the probability of available states Y can be calculated. By the Bayes rule, $P(Y \mid X)$ is represented as $P(X \mid Y)P(Y)/P(X)$. However, because the Naïve Bayesian Classifier wants to find the largest probability for Y, only $P(X \mid Y)P(Y)$ is considered.

$$\begin{aligned} P(Y \mid X) &= P(x_1, x_2, \ldots, x_n \mid Y)\, P(Y) \\ &= P(x_1 \mid Y)\, P(x_2 \mid Y) \cdots P(x_n \mid Y) \\ &= \prod_{i=1}^{n} P(x_i \mid Y). \end{aligned} \tag{2}$$

Then, the Naïve Bayesian Classifier becomes as

$$\begin{aligned} v &= \arg\max_{Y} P(X \mid Y)\, P(Y) \\ &= \arg\max_{Y} \prod_{i=1}^{n} P(x_i \mid Y)\, P(Y). \end{aligned} \tag{3}$$

At first, the proposed network access control method estimates the network status of WiFi through the inference engine. The inference engine employs the Naïve Bayesian Classifier of (3) and exploits successful transmission rate (x_1) and strength of reception signal (x_2) as attributes. State variable Y for WiFi has 0 and 1 as its value. $Y = 0$ means the network status of WiFi is bad and $Y = 1$ means the network status of WiFi is good. If there exists m training set as a priori

statistic information, the probabilities for WiFi state Y in the inference engine can be represented as

$$P(x_i \mid Y=1) = \frac{\sum_{j=1}^{m} 1\left\{x_i^{(j)}=1,\ y^{(j)}=1\right\}}{\sum_{j=1}^{m} 1\left\{y^{(j)}=1\right\}}, \quad (4)$$

$$P(Y=1) = \frac{\sum_{j=1}^{m} 1\left\{y^{(j)}=1\right\}}{m},$$

$$P(x_i \mid Y=0) = \frac{\sum_{j=1}^{m} 1\left\{x_i^{(j)}=1,\ y^{(j)}=0\right\}}{\sum_{j=1}^{m} 1\left\{y^{(j)}=0\right\}}, \quad (5)$$

$$P(Y=0) = \frac{\sum_{j=1}^{m} 1\left\{y^{(j)}=0\right\}}{m}.$$

Equation (4) represents the probability of WiFi with good status and (5) represents the probability of WiFi with bad status. The inference engine applies the probability values of (4) and (5) to (3) and then it estimates the WiFi status. In (4) and (5), the indicator function $1\{\cdot\}$ counts 1 if the given condition is satisfied.

If the inference engine has no a priori information, it cannot estimate the WiFi status because $P(X \mid Y)$ is 0. In this case, $P(Y \mid X)$ is also 0 so the network estimation cannot be performed. To avoid this case, the inference engine applies Laplace smoothing. The Laplace smoothing adds 1 to the numerator of (4) and (5) and adds k to the denominator of (4) and (5). The value k represents the number of states of Y. In this paper, because Y has 0 (bad) or 1 (good) as the WiFi status, k becomes 2. Then, when the Laplace smoothing is applied, (4) and (5) become

$$P(x_i \mid Y=1) = \frac{\sum_{j=1}^{m} 1\left\{x_i^{(j)}=1,\ y^{(j)}=1\right\}+1}{\sum_{j=1}^{m} 1\left\{y^{(j)}=1\right\}+2}, \quad (6)$$

$$P(Y=1) = \frac{\sum_{j=1}^{m} 1\left\{y^{(j)}=1\right\}+1}{m+2},$$

$$P(x_i \mid Y=0) = \frac{\sum_{j=1}^{m} 1\left\{x_i^{(j)}=1,\ y^{(j)}=0\right\}+1}{\sum_{j=1}^{m} 1\left\{y^{(j)}=0\right\}+2},$$

$$P(Y=0) = \frac{\sum_{j=1}^{m} 1\left\{y^{(j)}=0\right\}+1}{m+2}.$$

From (3) and (6), the WiFi network status can be estimated and the status value of the WiFi is used as an input parameter for the decision engine. The decision engine performs network access control using the estimated network status value and other system parameters. Table 1 shows an example of training data for learning. The inference engine maintains transmission rate and signal strength as training data and obtains the most possible probability using the maintained data.

3.2. Wireless Network Selection in the Decision Engine. The decision engine determines behaviors for the network selection using several statuses information such as WiFi status (y_1), buffer status (y_2), WiFi stability (y_3), and system status (y_4), which are as shown in Figure 2. It considers whether WiFi is good or not, data in the buffer is sufficient or not, WiFi is stable or not, and reception data rate for the network is greater than data consumption rate in the buffer or not.

Table 1: Training examples to predict network status.

APs	Transmission rate	Signal strength	Network status
1	120 kbps	−80 dBm	Bad
1	125 kbps	−83 dBm	Bad
1	110 kbps	−81 dBm	Bad
1	131 kbps	−78 dBm	Bad
2	2431 kbps	−52 dBm	Good
2	2105 kbps	−48 dBm	Good
2	2254 kbps	−56 dBm	Good

The decision engine controls the network access through the behavior decision table and it is represented in Algorithms 1 and 2. The WiFi status is estimated by the inference engine and the buffer status is obtained from a media player. In the proposed method, the buffer status is defined by three steps: high, normal, and low. The thresholds for the buffer of each step are represented as THRD_H, THRD_M, and THRD_L. The network stability is determined by the history of the network status. The decision engine manages history of the connected WiFi networks. If the ratio of good status is greater than the ratio of bad status in the history information, the network can be considered as stable. Otherwise, the network is considered as unstable. The system status is described with the variables, μ and λ. μ is content bit rate of a media player. λ is reception data rate of the connected network. The buffer is filled according to the data rate and the media player consumes data in the buffer according to the content bit rate.

In the proposed method, mobile terminals control the network access according to Algorithms 1 and 2 when they are connected to a WiFi network. Mobile terminals only connect to cellular network without any WiFi connections and they search for available WiFi networks and try to connect to them.

When the connected WiFi is good, if the amount of data in the buffer is sufficient (high step), the decision engine does not perform any operations. However, if the amount of data in the buffer is insufficient, the network access control will be performed according to stability and system status. In normal step of the buffer, mobile terminals search for another WiFi and try to migrate to the WiFi network. In low step of the buffer, if the WiFi is stable and μ is less than or equal to λ, it means incoming data in the buffer is greater than or equal to outgoing data. Thus, mobile terminals simultaneously use both cellular and WiFi until the amount of data in the buffer satisfies THRD_M. If μ is greater than λ, it means incoming data in the buffer is less than outgoing data. Thus, mobile terminals simultaneously use both cellular and WiFi until the amount of data in the buffer satisfies THRD_H and then the mobile terminals search for another WiFi and migrate to the WiFi network. If WiFi is unstable in low step of the

```
Decision-For-Network-Access-Control (y1, y2, y3, y4)
(1) μ ← y4 · μ
(2) λ ← y4 · λ
(3) If y1 is good then
(4)   If y2 is normal then
(5)     If y3 is stable then
(6)       If μ > λ then
(7)         Find another WiFi
(8)       End if
(9)     Else
(10)      If μ > λ then
(11)        Find another WiFi and then migrate to the WiFi
(12)      End if
(13)    End if
(14)  Else if y2 is low then
(15)    If y3 is stable then
(16)      If μ ≤ λ then
(17)        Use cellular and WiFi simultaneously until the buffer meets THRD_M
(18)      Else
(19)        Use cellular and WiFi simultaneously until the buffer meets THRD_H
(20)        Find another WiFi and then migrate to the WiFi
(21)      End if
(22)    Else
(23)      If μ ≤ λ then
(24)        Use cellular and WiFi simultaneously until the buffer meets THRD_H
(25)        Find another WiFi and then migrate to the WiFi
(26)      Else
(27)        Find another WiFi and then migrate to the WiFi
(28)        Or transfer to cellular
(29)      End if
(30)    End if
(31)  End if
(32) End if
```

Algorithm 1: Decision algorithm for network access control: good WiFi status.

buffer, when μ is less than or equal to λ, mobile terminals simultaneously use both cellular and WiFi until the amount of data in the buffer satisfies THRD_H and then the mobile terminals search for another WiFi and migrates to the WiFi network. When μ is greater than λ, the mobile terminal only uses cellular network or finds another WiFi network to migrate to it.

In case that WiFi status is bad and the amount of data in the buffer is sufficient, mobile terminals keep the current WiFi connection or search for another WiFi network according to the network stability. If the amount of data in the buffer is an intermediate level and the connected WiFi is stable, mobile terminals search for another WiFi network or migrate to the discovered WiFi network according to the system status. If WiFi is unstable and the reception data rate for data transmission (λ) is greater than or equal to the content bit rate (μ), mobile terminals simultaneously use both cellular and WiFi to fill the buffer until THRD_M. In case that the reception data rate is less than the content bit rate, mobile terminals simultaneously use both cellular and WiFi to fill the buffer until THRD_H. When the amount of data in the buffer is insufficient and the connected WiFi is stable, if the reception data rate is greater than or equal to the content bit rate, mobile terminals simultaneously use both cellular and WiFi to fill the buffer until THRD_H. If the reception data rate is less than the content bit rate, mobile terminals try to migrate to another WiFi network or transfer to cellular network. In case that the connected WiFi is unstable and the reception data rate is greater than or equal to the content bit rate, mobile terminals also try to migrate to another WiFi network or transfer to cellular network. However, if the reception data rate is less than the content bit rate, the mobile terminals do not use WiFi network. They just transfer to cellular network.

4. Performance Evaluation

4.1. Network Model. Mobile terminals include multiple network interfaces to access cellular and WiFi network. They move according to the network model as shown in Figure 3. That is, the mobile terminals move from cellular area to WiFi1 area and then they move from WiFi1 area to WiFi2 area. In WiFi2 area, the mobile terminals move to cellular area. The mobile terminals continuously move to these network areas. Each WiFi area is maintained for VISITING_TIME and the cellular area is maintained for VISITING_TIME/2. In this

```
Decision-For-Network-Access-Control (y1, y2, y3, y4)
(1) μ ← y4 · μ
(2) λ ← y4 · λ
(3) If y1 is bad then
(4)   If y2 is high then
(5)     If y3 is unstable then
(6)       Find another WiFi
(7)     End if
(8)   Else if y2 is normal then
(9)     If y3 is stable then
(10)      If μ ≤ λ then
(11)       Find another WiFi
(12)      Else
(13)        Find another WiFi and then migrate to the WiFi
(14)      End if
(15)    Else
(16)      If μ ≤ λ then
(17)        Use cellular and WiFi simultaneously until the buffer meets THRD_M
(18)      Else
(19)        Use cellular and WiFi simultaneously until the buffer meets THRD_H
(20)      End if
(21)    End if
(22)  Else if y2 is low then
(23)    If y3 is stable then
(24)      If μ ≤ λ then
(25)        Use cellular and WiFi simultaneously until the buffer meets THRD_H
(26)      Else
(27)        Find another WiFi and then migrate to the WiFi
(28)        Or transfer to cellular
(29)      End if
(30)    Else
(31)      If μ ≤ λ then
(32)        Find another WiFi and then migrate to the WiFi
(33)        Or transfer to cellular
(34)      Else
(35)        Transfer to cellular
(36)      End if
(37)    End if
(38)  End if
(39) End if
```

Algorithm 2: Decision algorithm for network access control: bad WiFi status.

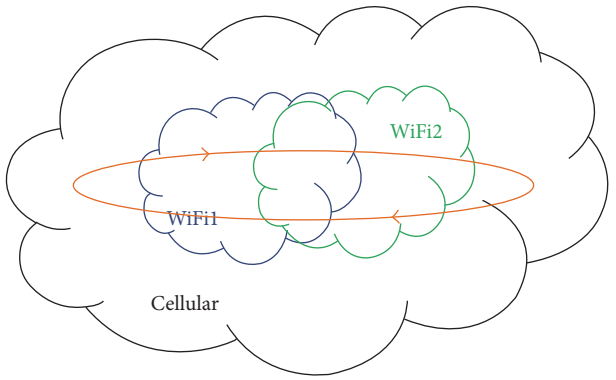


Figure 3: Network model.

network model, the mobile terminals monitor the status of the connected network and periodically perform the network access control.

4.2. Channel Model. The status of wireless channel for WiFi network can be divided into two categories which are denoted as good and bad. The status change of wireless channel can be modeled by Markov Chain. When probability of the status change from good to bad is p and probability of the status change from bad to good is q, the wireless channel can be modeled as Figure 4.

Then, Figure 3 is represented as

$$P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}. \tag{7}$$

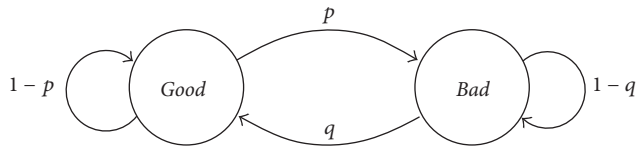


Figure 4: WiFi channel model.

When limiting probability [24] is applied to the Markov channel model, the probabilities of good or bad status can be obtained. The probabilities of the good and bad status are represented as

$$P\{X = \text{Good}\} = \frac{q}{p+q}, \quad P\{X = \text{Bad}\} = \frac{p}{p+q}. \tag{8}$$

Then, the probability of successful data transmission is represented as

$$P_s = (1 - \text{PER})\, P\{X = \text{Good}\}, \tag{9}$$

where the Packet Error Rate (PER) for the WiFi network is set to 1%.

In general, signal strength in WiFi networks is frequently changed due to channel environments such as interference. However, cellular network provides better wireless conditions. Thus, in this paper, the channel condition of cellular network is always assumed as good.

4.3. Simulation Environments. The simulator for performance evaluation is implemented with the SMPL library [25]. It is an event-driven simulation library using C language. The period of simulation is set to 50000 sec. A mobile terminal moves according to the network model in Section 4.1 and data rates of the wireless networks are set to 10 Mbps in cellular, 1 Mbps in WiFi1, and 5 Mbps in WiFi2. The VISITING_TIME in the simulation is set to 300 sec and the WiFi channel is changed per 120 sec according to the channel model in Section 4.2.

In WiFi2, the channel parameter p follows the uniform distribution between 0.1 and 0.2 and the parameter q follows the uniform distribution between 0.7 and 0.9. Signal strength of the WiFi network is randomly determined between −65 and −45 in good channel condition. In bad channel condition, the signal strength is randomly determined between −85 and −65. In WiFi1, the parameter p follows the uniform distribution between 0.2 and 0.4 and the parameter q follows the uniform distribution between 0.7 and 0.8. In good channel condition, the signal strength is randomly determined between −75 and −55. In bad channel condition, the signal strength is randomly determined between −90 and −75. In the WiFi networks, the network condition is assumed as good if P_s is greater than 0.8. If P_s is less than 0.7, the network condition is assumed as bad.

When the mobile terminal is in the WiFi networks, it monitors network per 5 sec and adds the training set (i.e., network information) of the network. At that time, the mobile terminal estimates the network status using the inference engine and then performs network access control using the decision engine. For WiFi networks, if the ratio of good channel is greater than the ratio of bad channel, it is assumed the network is stable. Otherwise, the network is assumed as unstable.

Table 2: Simulation parameters.

Parameters	Value
VISITING_TIME	300 sec
WiFi1 data rate	1 Mbps
WiFi2 data rate	5 Mbps
Cellular data rate	10 Mbps
CHANNEL_INTERVAL	120 sec
NET_MONITOR_INTERVAL	5 sec
THRD_H	6 MB
THRD_M	4 MB
THRD_L	1/1.5/2 MB
BUFFER_SIZE	8 MB
SIMULATION_TIME	50000 sec

When the mobile terminal moves, the streaming service is provided for the mobile terminal. A media player of the mobile terminal has 2 Mbps as CONTENT_RATE and data in the buffer is consumed according to the rate. The buffer size is 8 MB. If the whole buffer is filled with data, the mobile terminal will not request data to the media server. If the buffer is empty, the streaming service cannot be provided and the media player is in PAUSE status. The buffer parameter THRD_H is set to 6 MB and THRD_M is set to 4 MB. THRD_L is set to 1 MB, 1.5 MB, and 2 MB, respectively. Table 2 represents the simulation parameters for performance evaluation.

The proposed method is compared with the buffer-based network access control and no network access control. The buffer-based network access control is classified by two cases. (1) In the first case, network switching between WiFi and cellular networks occurs according to the buffer status. If the amount of data in the buffer is less than or equal to THRD_L, the mobile terminal releases the connection of WiFi and uses the only cellular network. If the amount of data in the buffer is greater than THRD_L, the mobile terminal tries to access the available WiFi networks. (2) In the second case, bandwidth aggregation between WiFi and cellular networks occurs according to the buffer status. If the amount of data in the buffer is less than or equal to THRD_M, the mobile terminal simultaneously exploits both cellular and WiFi networks. If the amount of data in the buffer is greater than THRD_H, the mobile terminal only uses WiFi network.

4.4. Simulation Results. For performance evaluation, the proposed method is compared to buffer-based network access control methods and no network access control. The pause counts of a media player in the streaming service and the amount of data traffic usage in WiFi and cellular network are measured through the computer simulation. Table 3 and

Table 3: Simulation result: pause counts.

	Proposed	Buffer 1	Buffer 2	None
Number of pauses	0	0	0	17109

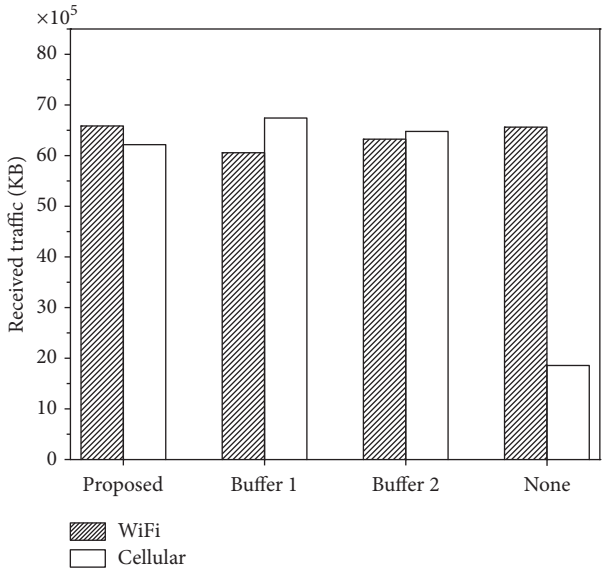


Figure 5: Simulation result: received traffic at the mobile terminal.

Table 4: Simulation result according to THRD_L: pause counts.

	1 MB	1.5 MB	2 MB
Number of pauses	103	4	0

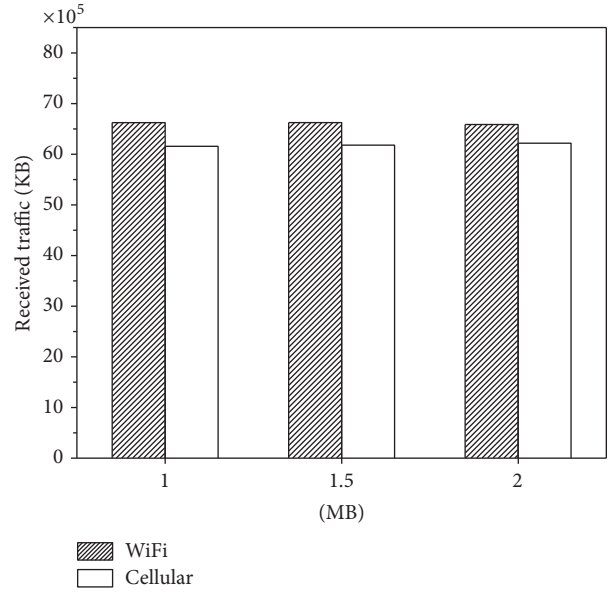


Figure 6: Simulation result according to THRD_L: received traffic at the mobile terminal.

Figure 5 show the simulation results when the lowest buffer threshold (THRD_L) is 2 MB.

Table 3 represents pause counts and Figure 5 shows received traffic through cellular and WiFi networks at the mobile terminal. In Table 3 and Figure 5, "Proposed" indicates the proposed method and "None" indicates no network access control. "Buffer 1" is the first case of the buffer-based method and "Buffer 2" is the second case of the buffer-based method. As shown in Table 3, by applying the network access control, the pause counts of the media player can be largely reduced. As shown in Figure 5, when the network access control is applied, received traffic of cellular network is greatly increased. By exploiting the cellular network when the WiFi is not good, usage of the cellular network is increased while the user experience is improved. That is, users can be served seamless streaming services without any pauses. The "Buffer 1" method disconnects from WiFi and uses the cellular network when the WiFi is bad and the amount of data in the buffer is not sufficient. In this case, the amount of traffic usage of the cellular network is the largest. The "Buffer 2" method exploits both the WiFi and the cellular network when the WiFi is bad and the amount of data in the buffer meets THRD_M. Thus, it can increase usage of WiFi network. However, the proposed method considers more factors to access wireless networks. The proposed method includes buffer status, WiFi status, network stability, data rate of networks, and content bit rate of the media player. Through the proposed method, usage of WiFi networks is maximized and usage of cellular network is minimized while seamless streaming service is provided. According to the results, it is verified that the proposed method is more efficient in aspect of data offloading (data offloading means usage of complementary networks such as WiFi instead of cellular networks). Data offloading rate of "Proposed" is 51%. "Buffer1" and "Buffer2" are 47% and 49%, respectively.

Table 4 represents pause counts and Figure 6 shows received traffic through cellular and WiFi networks at the mobile terminal according to the lowest threshold of the buffer (THRD_L). The proposed method selects different behaviors in the network access control according to the buffer status. In the decision engine, the proposed method has several actions according to the parameters when the amount of data in the buffer is lower than the threshold, THRD_L. Thus, the results according to the change of the lowest threshold of the buffer are shown in Table 4 and Figure 6.

When the lowest threshold of the buffer is reduced, pause counts of the media player are increased as shown in Table 4. The difference of WiFi traffic and cellular traffic is increased as shown in Figure 6: the differences are 467,456 KB, 445,440 KB, and 369,664 KB when THRD_L is 1 MB, 1.5 MB, and 2 MB, respectively. That is, the WiFi traffic is increased and the cellular traffic is decreased. The reason is that the behaviors for network access control are lately performed when the amount of data in the buffer meets the lowest threshold of the buffer under the bad WiFi condition. Thus, more pause counts occur when the lowest threshold is reduced: the pause counts are 103, 4, and 0 when THRD_L is 1 MB, 1.5 MB, and 2 MB, respectively. Therefore, the threshold

value should be determined according to the goal of the designed system. If the goal of the system is to maximize WiFi usage, the lowest threshold of the buffer should be minimized and if the goal of the system is to improve user experiences, the threshold value should be increased. By adjusting the threshold value according to the system goal, network access control can be tuned.

As shown in Table 3, Figure 5, Table 4, and Figure 6, if the mobile terminal does not perform the network access control in heterogeneous wireless networks, good user experiences cannot be provided. In addition, the network access control by considering only the buffer status is difficult to control the network access accurately. Thus, as the proposed method, various elements should be considered for the efficient network access control.

5. Conclusion

Mobile data services provide various business opportunities. The location-based mobile data services are increased. Particularly, the growth of the location-based video services is expected. In addition, recent advances in mobile applications have enabled mobile terminals to use multiple network interfaces. Mobile operating systems support controlling an individual network interface. Application protocol such as HTTP Range Requests provides transmitting a data block as a chunk in total data size. Therefore, network access control is available. Controlling network access according to the network status affects quality of services of the mobile applications. In case of the mobile terminal such as smartphones, applications provide services over both cellular and WiFi networks. The cellular network provides stable network connectivity in wide area. However, the cellular network takes high costs. Although the WiFi network takes no costs, it has limitation of the network coverage and its network status is frequently changed. In addition, video traffic has the largest portion in total mobile traffic. If users are served video streaming services while they are moving, maintaining the quality of services is very important. However, the quality of services is not guaranteed in the only WiFi network. Therefore, it is significant that the mobile terminal performs the network access control in heterogeneous wireless networks according to the network status.

In general, the network access control is based on the amount of data in the buffer but this case is difficult to accurately control the network access. The proposed method performs the network access control by considering the network status such as current status and stability by the network history. It also reflects the buffer status to decide the behaviors for the network access control. Through the inference engine based on Naïve Bayesian Classifier, the proposed method estimates the WiFi network status. The estimated results (i.e., the history information of WiFi networks) and the buffer status are reflected to decide the behaviors for the network access control. Therefore, the proposed method can provide the efficient network access. As the results, the proposed method can guarantee quality of the seamless services. In addition, because the proposed method increases the usage of the WiFi networks, the costs for network usage can be reduced. Although this paper uses the fixed buffer parameters, the parameters can be varied according to network status. This point can be applied to the proposed approach as future works.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1D1A1B03931406), and this work was supported by the Soonchunhyang University Research Fund.

References

[1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast, 2015–2020," 2016.

[2] J. Summers, T. Brecht, D. Eager, and B. Wong, "To chunk or not to chunk: implications for HTTP streaming video server performance," in *Proceedings of the 22nd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '12)*, Ontario, Canada, 2012.

[3] M. Jang, H. Oh, J. Yang, J. K. Choi, K. Kim, and I. Cho, "Implementation of continuous HTTP live streaming using playback position request mechanism in heterogeneous network," in *Proceedings of the International Conference on Advanced Communication Technology (ICACT '13)*, pp. 990–993, Daejeon, Korea, 2013.

[4] R. Fielding, Y. Lafon, and J. Reschke, Hypertext transfer protocol (HTTP/1.1): range requests," IETF RFC 7233, 2014.

[5] D. Yun and K. Chung, "Rate adaptation for HTTP video streaming to improve the QoE in multi-client environments," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 11, pp. 4519–4533, 2015.

[6] H. S. Kim, I. Kim, K. Han, D. Kim, J. S. Seo, and M. Kang, "An adaptive buffering method for practical HTTP live streaming on smart OTT STBs," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 3, pp. 1416–1428, 2016.

[7] A. Biernacki, "Server side solutions for web-based video," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 4, pp. 1768–1789, 2016.

[8] D. H. Bui, K. Lee, S. Oh et al., "GreenBag: energy-efficient bandwidth aggregation for real-time streaming in heterogeneous mobile wireless networks," in *Proceedings of the IEEE Real-Time Systems Symposium (RTSS '13)*, pp. 57–67, 2013.

[9] S. Nirjon, A. Nicoara, C.-H. Hsu, J. P. Singh, and J. A. Stankovic, "MultiNets: a system for real-time switching between multiple network interfaces on mobile devices," *ACM Transactions on Embedded Computing Systems*, vol. 13, no. 4, article 121, 2014.

[10] X. Cai, L. Chen, R. Sofia, and Y. Wu, "Dynamic and user–centric network selection in heterogeneous networks," in *Proceedings of the IEEE International Performance, Computing and Communications Conference*, pp. 538–544, 2007.

[11] A. Aijaz, H. Aghvami, and M. Amani, "A survey on mobile data offloading: technical and business perspectives," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 104–112, 2013.

[12] S. Qutub and T. Anjali, "Dynamic implementation of network selection policies," in *Proceedings of IEEE Conference on Local Computer Networks (LCN '10)*, pp. 292–295, 2010.

[13] M. M. Alkhawlani and A. M. Mohsen, "Hybrid approach for radio network selection in heterogeneous wireless networks," *International Journal of Advanced Science and Technology*, vol. 44, 2012.

[14] L. Wu and K. Sandrasegaran, "A survey on common radio resource management," in *Proceedings of the International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless '07)*, 2007.

[15] A. Hasib and A. O. Fapojuwo, "Analysis of common radio resource management scheme for end-to-end QoS support in multiservice heterogeneous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4, pp. 2426–2439, 2008.

[16] C. Shin, J. Cho, J. G. Kim, and B. Lee, "An AHP-based resource management scheme for CRRM in heterogeneous wireless networks," *Annals of Telecommunications*, vol. 67, no. 11-12, pp. 511–522, 2012.

[17] S. Ahmed and S. S. Kanhere, "A Bayesian routing framework for delay tolerant networks," in *Proceedings of the the IEEE Wireless Communications and Networking Conference (WCNC '10)*, pp. 1–6, 2010.

[18] S. Marsland, *Machine Learning an Algorithmic Perspective*, Chapman and Hall, New York, NY, USA, 2009.

[19] E. F. Flushing, J. Nagi, and G. A. Di Caro, "A mobility-assisted protocol for supervised learning of link quality estimates in wireless networks," in *Proceedings of International Conference on Computing, Networking and Communications (ICNC '12)*, pp. 137–14152, 2012.

[20] I. El Khayat, P. Geurts, and G. Leduc, "Enhancement of TCP over wired/wireless networks with packet loss classifiers inferred by supervised learning," *Wireless Networks*, vol. 16, no. 2, pp. 273–290, 2010.

[21] M. A. Alsheikh, S. Lin, D. Niyato, and H. P. Tan, "Machine learning in wireless sensor networks: algorithms, strategies, and applications," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 4, pp. 1996–2018, 2014.

[22] M. Di and E. M. Joo, "A survey of machine learning in wireless sensor networks from networking and application perspectives," in *Proceedings of International Conference on Information, Communications Signal Processing*, 2007.

[23] D. Kim, Y. Jeong, and S. Kim, "Data-filtering system to avoid total data distortion in iot networking," *MDPI Symmetry Journal*, vol. 9, no. 1, article 16, 2017.

[24] S. M. Ross, *Probability Models for Computer Science*, Harcourt/Academic Press, 2001.

[25] M. H. MacDougall, *Simulating Computer Systems, Techniques and Tool*, MIT Press, 1987.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 8429878, 8 pages
https://doi.org/10.1155/2017/8429878

Research Article

A Traffic Prediction Model for Self-Adapting Routing Overlay Network in Publish/Subscribe System

Meng Chi,[1] Jianhua Yang,[1,2] Yabo Liu,[1] and Zhenhui Li[3]

[1]*College of Computer Science and Technology, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang 310027, China*
[2]*The Sci-Tech Academy, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang 310027, China*
[3]*College of Control Science and Engineering, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang 310027, China*

Correspondence should be addressed to Jianhua Yang; jhyang@zju.edu.cn

Received 20 January 2017; Accepted 27 February 2017; Published 30 March 2017

Academic Editor: Jaegeol Yim

In large-scale location-based service, an ideal situation is that self-adapting routing strategies use future traffic data as input to generate a topology which could adapt to the changing traffic well. In the paper, we propose a traffic prediction model for the broker in publish/subscribe system, which can predict the traffic of the link in future by neural network. We first introduced our traffic prediction model and then described the model integration. Finally, the experimental results show that our traffic prediction model could predict the traffic of link well.

1. Introduction

Location-based services (LBS) have drawn more and more attention, which can provide us with location-aware experiences. Some LBS applications such as E-coupon and Mobile Buddy List were implemented based on publish/subscribe system.

Providing the fast LBS service to millions of users is a big challenge; self-adapting overlay network and reducing traffic based on traffic prediction in publish/subscribe system provide the solution to this problem, which could deliver a message to end user efficiently.

There are two improvements which will be achieved when the underlying infrastructure of the system is incorporated with publish/subscribe paradigm. One is that this paradigm will provide anonymous communication mechanism. The other is that this paradigm will decouple consumers and producers in terms of space, time, and synchronization [1]. As a result, the publish/subscribe paradigm has been researched both in academia and in industry recently.

In publish/subscribe system, producers send notifications, and consumers receive their interested notifications expressed by the subscriptions [2]. The overlay network forwards notifications to consumers according to the match and routing algorithm.

The content-based publish/subscribe system is commonly applied in many scenarios, and its cost of routing depends on the topology of the dispatching network which is usually defined at deployment time and never changes [3, 4], so it is important to reconfigure the dispatching network at runtime to reduce the overall routing cost. There is always an assumption in the current self-adapting routing strategies [3, 5, 6]; the assumption is that the traffic of the link l at time t is equal to traffic of the link l at time $t + 1$:

$$v_t^l = v_{t+1}^l. \tag{1}$$

Ideally, if a self-adapting routing strategy could reduce the traffic cost of the overlay network efficiently, v_{t+1}^l should be used as input data to figure out the appropriate dispatching network at time t, but actually it cannot get v_{t+1}^l at time t, so it could only assume that $v_t^l = v_{t+1}^l$. However, in fact the traffic of the link is constantly changing, so $v_t^l \neq v_{t+1}^l$, which means that the traffic of the link at time t which is used to figure out the dispatching network is not real traffic of the link at time

$t + 1$. Consequently, the reconfigured overlay network cannot adapt well to the traffic at time $t + 1$.

In the paper, we propose a traffic prediction model which could predict the traffic of the overlay network to overcome the limitation above. We predict v_{t+1}^{l} at time t and neural network is used in our traffic prediction model to predict the traffic of the link. In this way, the self-adapting routing strategy could use the predicted traffic to reconfigure the dispatching network well.

The paper is organized as follows. Section 2 discusses related work. Section 3 proves that the traffic is predictable. Section 4 presents the traffic prediction model. Section 5 presents the model integration. Section 6 shows and discusses the experimental results. Section 7 presents our conclusion.

2. Related Work

Several approaches of self-adapting routing strategy and prediction have been researched in the past. We present and discuss these approaches as follows.

In 2014, we proposed a self-adapting routing strategy for frequently changing application traffic in content-based publish/subscribe system and published in the literature [5]. The strategy firstly records the traffic information and then uses it to figure out a new topology. In the end, the strategy reconfigures the topology of the overlay network to reduce the overall traffic cost. Our research is based on assumption (1), so the reconfigured overlay network cannot adapt well to the traffic at time $t + 1$.

A self-organizing algorithm to solve the problem of publish/subscribe overlay decision problem (PSODP) is presented in the literature [6]. In another work [3], a distributed algorithm to solve the problem of optimal content-based routing (OCBR) was proposed. However, both of these two approaches also are based on assumption (1), so the reconfigured overlay network cannot adapt well to the traffic at time $t + 1$.

A review of neural networks for the prediction and forecasting of water resources variables was summarized in literature [7]. These include the choice of performance criteria, the division and preprocessing of the available data, the determination of appropriate model inputs and network architecture, optimization of the connection weights (training), and model validation. Also, the neural networks prediction theory is applied in other fields, such as traffic flow prediction and stock prediction [8, 9]. However, it is firstly introduced in publish/subscribe system.

3. Predictability Analysis

We shall first prove that the traffic of the publish/subscribe overlay network is predictable before we introduce our traffic prediction model.

In the overlay network, a broker node, connected by several links which contains the inputs and outputs, is our research object, and we want to predict the traffic of a certain link in the broker node. However, the traffic in the broker node always exhibits complicated, irregular behaviors which

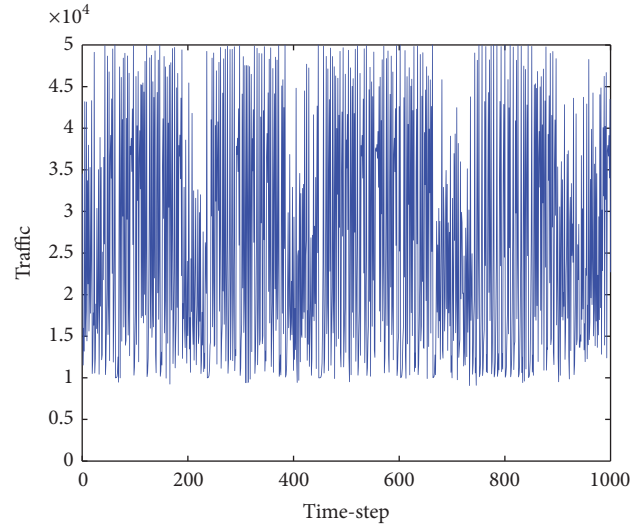


Figure 1: Traffic-time series.

are seemingly random, and they are largely determined by the business logic of the upper level, so we can come to the conclusion that it is impossible to make a prediction for the next hour or day traffic.

On the other hand, short-term (1–15 minutes) prediction has been analyzed as below. The data was recorded from an air traffic control system which is used to monitor the designated airplane by different terminals, and the communications layer of this system is supported by the publish/subscribe system. A node n was chosen randomly in inner brokers, and a link l was chosen randomly as well in node n. We set the node n to record the routing notifications in every 5 minutes, and we call this period as a time-step. We recorded notifications routed by link l 1000 time-steps in the node, which comprise a time series ts. The data are shown in Figure 2.

In Figure 1, x-coordinate is the number of the time-steps and the y-coordinate is the number of the notifications. From the figure, the curve does not show obvious periodicity or complete randomness, and it is still difficult to determine whether there is a certain rule in this time series ts. However, the subscriptions and the advertisement in the publish/subscribe system certainly have the prediction information, because they indicate what type of notification will be sent or received. To find this order and pattern, we introduced chaos theory into our proof procedure.

In chaos theory, chaos refers to an apparent lack of order in a system that nevertheless obeys particular laws or rules, and it is not disorder but a higher order of the universe [10, 11]. If the time series ts exhibits chaos, the traffic of link l could be predictable. The existence of a positive Lyapunov exponent is usually taken as an indication of the chaotic character [12]. In practice, we only need to calculate the largest Lyapunov exponent λ from this time series ts by small data sets method. If $\lambda > 0$, this time series ts is chaotic [13].

All calculation procedures will be done in MATLAB, and the input parameters of the largest Lyapunov exponent algorithm should be calculated firstly, which is shown as follows: reconstruction delay is 2 calculated by the fast

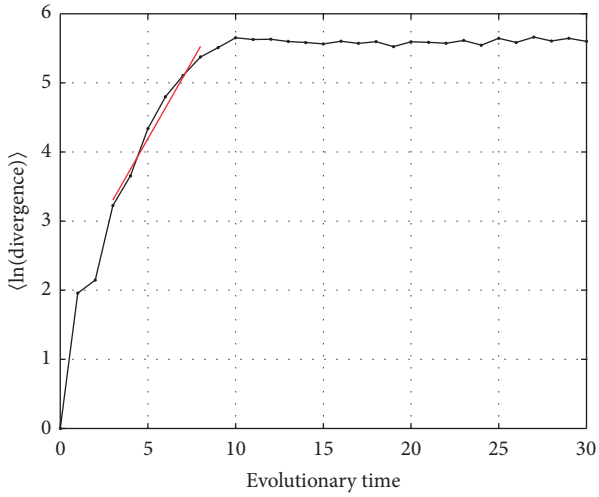


Figure 2: Largest Lyapunov exponent figure.

Fourier transform, mean period is 2 calculated by the fast Fourier transform, and embedding dimension is 3 calculated by C-C algorithm [14]. Now the largest Lyapunov exponent λ could be calculated, and the result is shown in Figure 2.

In Figure 2, x-coordinate is the evolutionary time and the y-coordinate is the $\langle \ln(\text{divergence}) \rangle$. The red line is calculated using least-squares fit method. The slope of this red line is the largest Lyapunov exponent λ. The slope is 0.4363, and the slope is larger than 0, so the time series ts exhibits chaos. We could arrive at the conclusion that the traffic of the publish/subscribe overlay network is predictable.

4. Publish/Subscribe Traffic Prediction Model

The publish/subscribe traffic prediction model (PSTPM) worked in content-based publish/subscribe system, and the advertisements mechanism is applied in the system. We assume that the subscription routing table and the advertisement routing table are updated by covering-based routing algorithm in the broker node.

In the publish/subscribe overlay network, a broker node is our modeling object, it is connected by several links which contains the inputs and outputs, and the traffic of a certain link in the node is our prediction object.

In Figure 3, if the traffic of the link L_k at time $t+1$ is our prediction object, we need to find out which relevant factors the traffic of the link L_k at time $t+1$ depends on. We consider these factors in two categories. One is the unchanged part which is relevant to the history traffic of link L_k, such as the traffic at time t, $t-1$ or $t-2$: here we use two history values $v_t^{L_k}$ and $v_{t-1}^{L_k}$ as the one part of the relevant factors. The other part is changed part which is the traffic that will increase or decrease in the link L_k at time $t+1$: if the subscription entry (S_i, D_i) was added to (deleted from) the subscription routing table at time t in node N_i and the destination D_i is node N_j, it denotes that the matching notifications have to (could not) be forwarded to the destination D_i through the link L_k at time $t+1$. If the advertisement entry (A_i, D_i) was added to (deleted from) the advertisement routing table at time t in node N_i and the destination D_i is node N_j, it denotes that the notifications might (could not) be received from the destination D_i through the link L_k at time $t+1$. Both cases will lead to traffic change in link L_k at time $t+1$, so the number of the change values $\text{sub}v_t^{L_k}$ and the number of the change values $\text{ad}v_t^{L_k}$ are the other two relevant factors. $\text{sub}v_t^{L_k}$ is the sum of forwarding notifications matched by S_i in node N_i and $\text{ad}v_t^{L_k}$ is the sum of forwarding notifications matched by A_i in node N_i.

$$\begin{aligned} \text{sub}v_t^{L_k} &= \sum_{j=1}^{n} \text{Num}\left(\text{Match}\left(\text{Notification}_j, S_i\right)\right), \\ \text{ad}v_t^{L_k} &= \sum_{j=1}^{n} \text{Num}\left(\text{Match}\left(\text{Notification}_j, A_i\right)\right). \end{aligned} \tag{2}$$

In formula (2), the function Num() returns the number of notifications; the function match() returns notification if the notification is matched by filter S_i or A_i; n is the length of the routing table.

In sum, we find out four factors: $v_t^{L_k}$, $v_{t-1}^{L_k}$, $\text{sub}v_t^{L_k}$, and $\text{ad}v_t^{L_k}$, and we use these factors to predict the traffic of link L_k $v_{t+1}^{L_k}$. We need to find some approaches to represent the relation between these two parts.

Neural networks have become extremely popular for prediction and forecasting in many areas [15, 16]. The advantage of neural networks lies in their ability to represent both linear and nonlinear relationships and in their ability to learn these relationships directly from the data being modeled. Now the neural network is used to model our problem and network parameters are discussed as below.

4.1. PSTPM Inputs and Outputs. As in any prediction model, the selection of appropriate model inputs and outputs is very important. In our prediction model, the inputs and outputs are determined by problem itself, so the inputs and outputs are discussed above.

Inputs: $v_t^{L_k}$, $v_{t-1}^{L_k}$, $\text{sub}v_t^{L_k}$, and $\text{ad}v_t^{L_k}$

Outputs: $v_{t+1}^{L_k}$

4.2. PSTPM Data Source. The prediction model is deployed in each broker node, and it is set to record the routing notifications in every t minutes and its default setting is 5 minutes. As a result, the recorded data comprise a time series, input data, and target data, which are used in training the network and could be figured out from the time series according to the selection of inputs and outputs.

4.3. PSTPM Data Division. Cross-validation technique [14] is used in our prediction model to divide the inputs data and targets data into three subsets: a training set which is used for training and accounted for 70%; a validation set which is used to validate that the network is generalizing and to stop training before overfitting and accounted for 15%; a testing

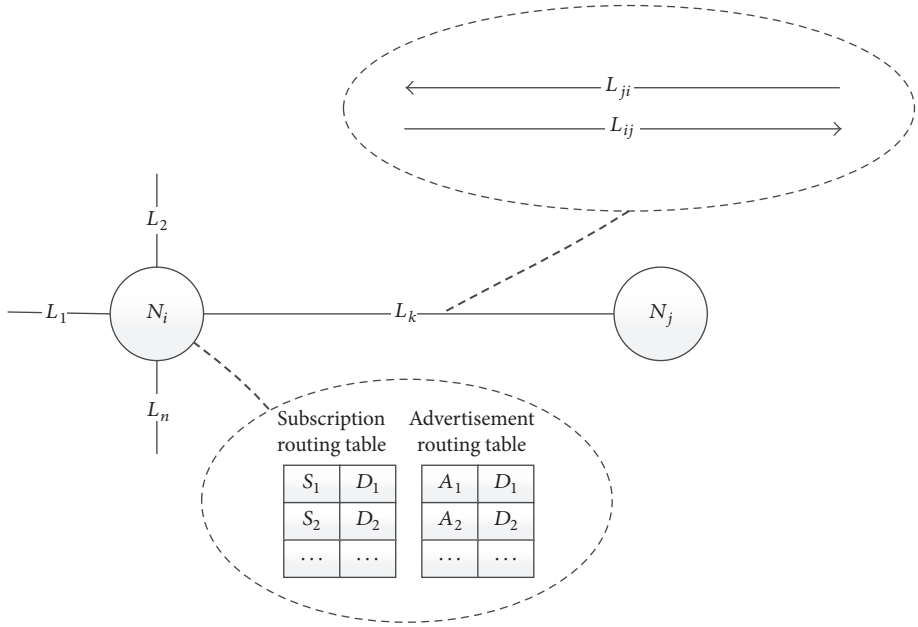


Figure 3: Broker node.

set which is used as a completely independent test of network generalization and accounted for 15%.

4.4. PSTPM Data Preprocessing. Generally, inputs data and targets data span different ranges. In order to ensure that all the data will be commensurate with the limits of the activation functions, in our prediction model, inputs data and targets data should be normalized to a value between 0 and 1 using formula (3).

$$V_t = \frac{v_t - v_{\min}}{v_{\max} - v_{\min}}. \quad (3)$$

In the formula, v_t is the sample data in the time series at time t. $v_{\max}$ and $v_{\min}$ are the maximum and minimum values in the time series. Obviously, the output results of the network which are between 0 and 1 should be carried out with reductions using formula (3).

4.5. PSTPM Network Structure. Feedforward network is used in our prediction model and it is arranged by three layers: an input layer, a hidden layer, and an output layer. The number of neurons in the input layer is fixed by the number of model inputs: 4; the number of neurons in the output layer equals the number of model outputs: 1; the number of neurons in hidden layer nodes was obtained using trial and error and the initial number of neurons is calculated by formula (4):

$$l = \sqrt{n + m} + a. \quad (4)$$

In formula (4), n is the number of the neurons in the input layer; m is the number of the neurons in the output layer; a is one number between 1 and 10.

In addition, we also consider the experience that the number of the hidden nodes in each layer should be between the size of input and output layer.

At last, the number of neurons in the hidden layer is 3.

4.6. PSTPM Network Optimization. In our prediction model, the back-propagation algorithm is used to train network, and the Levenberg-Marquardt (LM) optimization method is used to update the weight and bias. The initial weights are initialized to zero-mean random values in (−1,1). The Tan-Sigmoid transfer function is used in hidden layer and the Log-Sigmoid transfer function is used in output layer. The learning rate is set to 0.1. The error function is used as the mean squared error function, and it is set to 0.001. The maximum epoch size is set to 1000.

5. Model Integration

In this section, we show how the traffic prediction model integrates into the self-adapting routing strategy. Many approaches have been presented to solve the publish/subscribe overlay optimization problem [3, 5, 6]. Normally, the main idea of the self-adapting overlay routing strategy is to reduce the distance between brokers that consume a lot of identical notifications. In this process, the brokers need to know which links will forward a lot of the identical messages in the future, and they will reconfigure the topology of the overlay network to reduce the traffic cost according to these pieces of information. In other words, the more precise the prediction for the traffic that could be given by broker, the more the traffic cost reduction that will be achieved. Consequently, the prediction model presented in this paper could be applicable in all the strategies.

By the above analysis, we could know that the traffic prediction models are running in each broker node, and the model instances are created for each link. The working procedure figure is shown in Figure 4.

In Figure 4, the working procedure contains three procedures: training procedure, prediction procedure, and adapting procedure. In the training procedure, the brokers

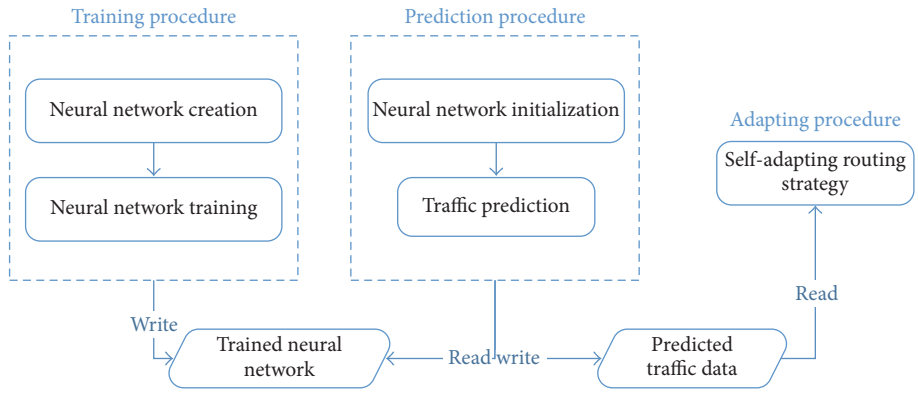


Figure 4: Working procedure.

create the neural network, train the network, and store the trained network for each link. The training procedure runs once a day. In the prediction procedure, the brokers predict the traffic for each link and save the predicted data. In the adapting procedure, the brokers reconfigure the topology of the overlay network using the predicted data. The brokers enter adapting procedure after finishing the prediction procedure; the prediction procedure is triggered after a certain time interval set by system administrator.

6. Experiments

In this section, our experiments were divided into two parts: one part is to validate the traffic prediction model and the other is to validate self-adapting strategy integrated into the prediction model.

For part one, we designed the experiments:

(i) To validate whether the prediction model proposed in this paper has the ability to predict the traffic

For part two, we designed the experiments:

(i) To validate whether the strategy integrated into the prediction model has the ability to reduce the traffic cost of the overlay network and to compare with other strategies

The part one experiments were simulated on MATLAB and the part two experiments were simulated on ProtoPeer [17] which is a distributed systems prototyping toolkit.

6.1. Part One Experiments. The part one experiments are to validate whether the prediction model proposed in this paper has the ability to predict the traffic. The data were recorded from the ProtoPeer environment and analyzed in MATLAB. We designed two experiments in this part: one is inner broker case where the traffic is relatively large and the other one is the border case where the traffic is relatively small. The experiment process is as follows.

A node n_{inn} was chosen randomly in inner brokers and a link l_{inn} was chosen randomly in node n_{inn}. We recorded notifications routed by link l_{inn} 1000 time-steps. The first few

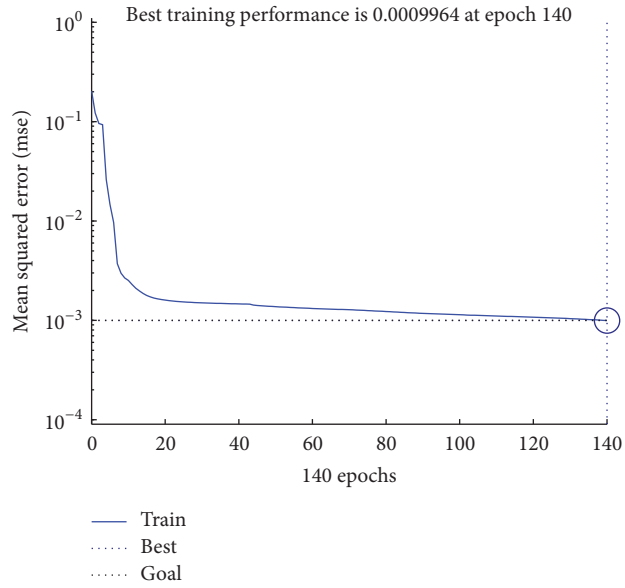


Figure 5: l_{inn} performance.

950 time-steps were used for training the network and the rest were used to test the network. The records were loaded and trained in MATLAB. The training performance figure is shown in Figure 5.

In Figure 5, x-coordinate is the number of the epochs and the y-coordinate is the mse. The figure shows that the mse reached 0.0009964 at iteration of 140 epochs, so the convergence speed of the error is very fast with LM algorithm. The regression figure is shown in Figure 6.

In Figure 6, x-coordinate is the targets and the y-coordinate is the network outputs. The best linear fit is indicated by a dashed line. The perfect fit is indicated by the solid line. In this figure, we cannot find out the dashed line and the solid line because they covered by the data points which composed a line. This means that the fit is very good. The results of the prediction are shown in Figure 7.

In Figure 7, x-coordinate is the number of the time-steps and the y-coordinate is the number of the notifications. The symbol "∘" denotes the actual value and the symbol "+"

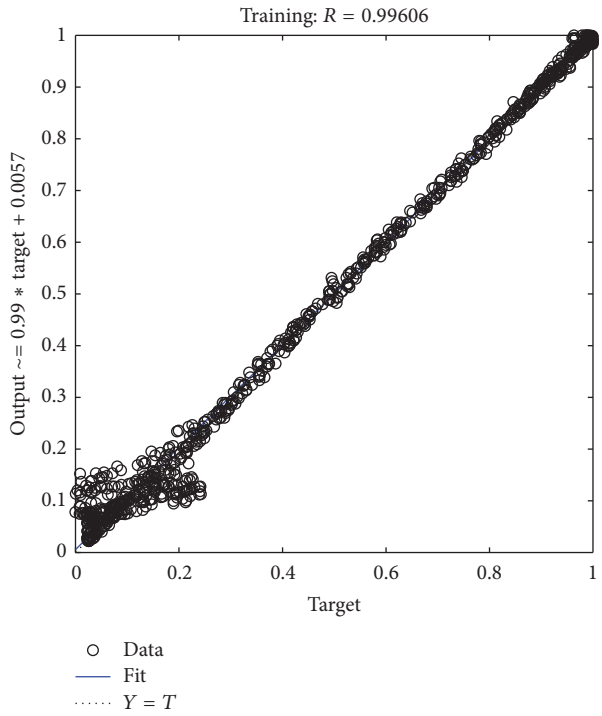


Figure 6: l_{inn} regression.

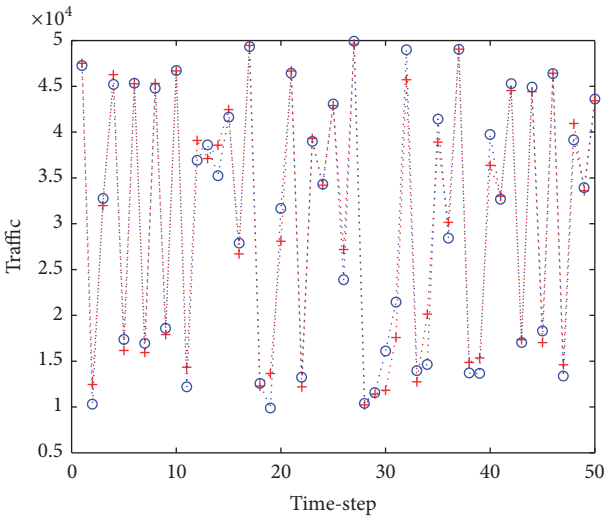


Figure 7: l_{inn} prediction.

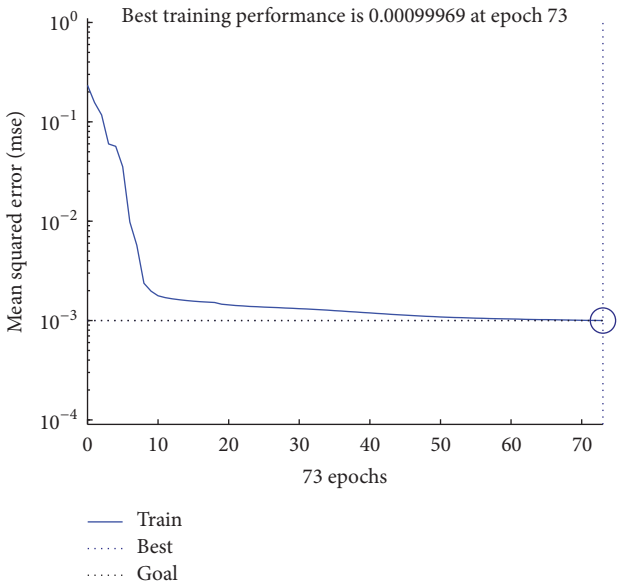


Figure 8: l_{brd} performance.

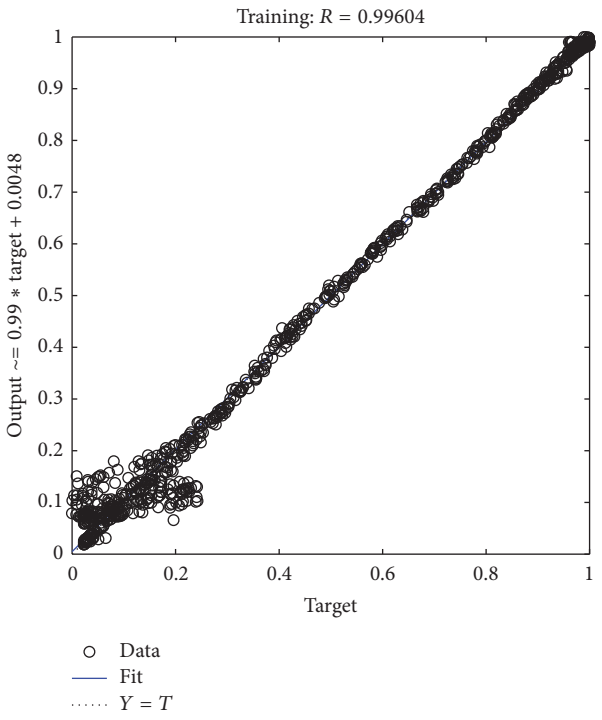


Figure 9: l_{brd} regression.

denotes the predicted value. From the figure, we can see that the predicted values series could well fit the actual ones. Next, border broker case was also given.

A node n_{brd} was chosen randomly in border brokers [18] and a link l_{brd} was chosen randomly in node n_{brd}. The rest of the environment is identical with the previous. The training performance figure is shown in Figure 8.

In Figure 8, the figure shows that the mse reached 0.00099969 at iteration of 73 epochs, so the convergence speed of the error is also very fast in border broker case. The regression figure is shown in Figure 9.

In Figure 9, the fit is also very good in border broker case. The results of the prediction are shown in Figure 10.

In Figure 10, we can see that the predicted values series could well fit the actual ones in border broker case. As a result, we can come to the conclusion that the prediction model proposed in this paper has the ability to predict the traffic.

6.2. Part Two Experiments. The part two experiments are to validate whether the strategy integrated into the prediction

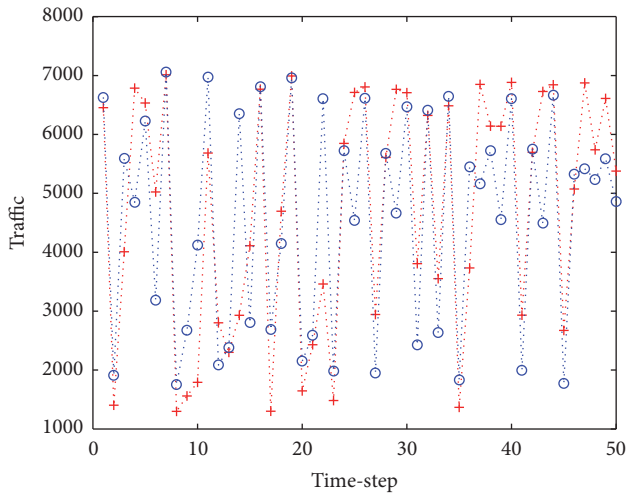


Figure 10: l_{brd} prediction.

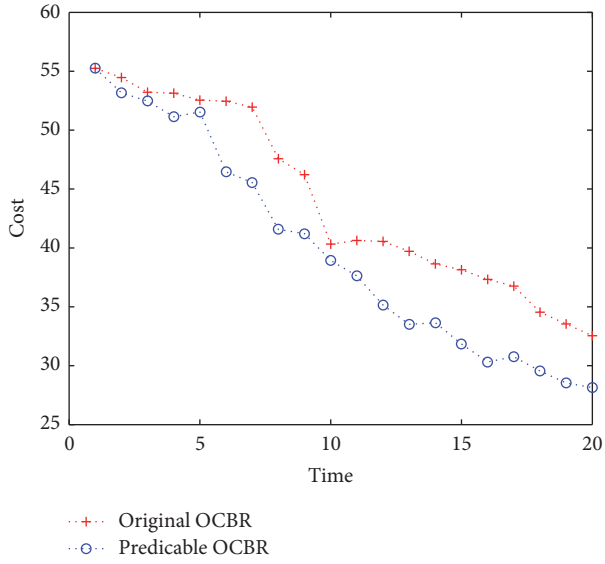


Figure 11: OCBR case.

model has the ability to reduce the traffic cost of the overlay network and to compare with other strategies. The traffic prediction model was implemented in ProtoPeer, and then the experiment analysis procedure was run in the ProtoPeer and the analysis results were recorded. Finally, the results were plotted in MATLAB. We designed two experiments in this part: one is PSOO-FCAT strategy and the other one is OCBR strategy. The strategies integrated into our traffic prediction model will be compared with the original ones. The experiment process and parameters are as follows.

An overlay network topology with 5000 nodes has been generated randomly, and 1000 nodes have been deployed on the brokers randomly. We randomly generated 2000 different types of subscription, 10000 different types of events based on the 1500 subscriptions, 8000 advertisements according to 8000 different types of events, 200 producers, 200 consumers, and 50 pairs of the ⟨producer, consumer⟩ that connected to brokers randomly, each producer published 40 types of events, each consumer subscribed to 8 types of events, and the event is sent every 2 simulation ticks by a producer. The simulation experiment was performed in 100 minutes. The results were recorded by us at every 500 ticks. The average value was calculated as shown in Figure 11.

In Figure 11, the original OCBR and the predicable OCBR cost are both decreased, but predicable OCBR cost decreases sharper than original OCBR, which indicates that both the original OCBR and the predicable OCBR strategy have the ability to reduce the traffic cost of the overlay network and the efficiency of the predicable OCBR strategy is more obvious.

In Figure 12, the original PSOO-FCAT and the predicable PSOO-FCAT cost are both decreased, but predicable PSOO-FCAT cost decreases sharper than original PSOO-FCAT, which indicates that both the original PSOO-FCAT and the predicable OCBR strategy have the ability to reduce the traffic cost of the overlay network and the efficiency of the predicable PSOO-FCAT strategy is more obvious.

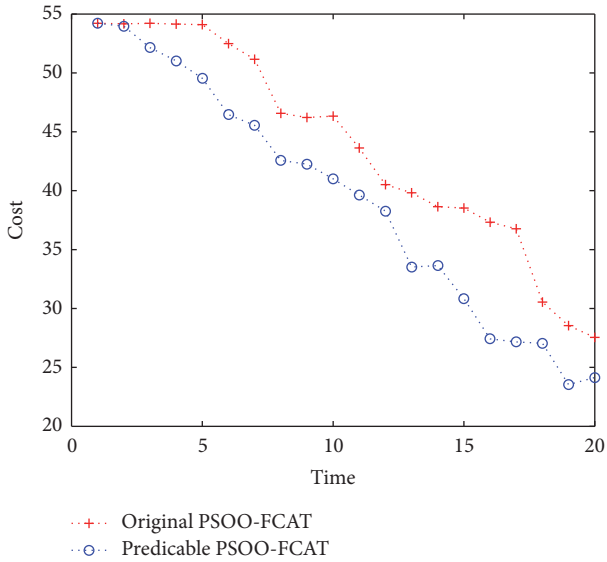


Figure 12: PSOO-FCAT case.

In sum, we can come to the conclusion that the strategy integrated into the prediction model has the ability to reduce the traffic cost of the overlay network.

7. Conclusion

In the paper, we propose a traffic prediction model for the broker in publish/subscribe system, and it uses neural network to predict the traffic of the link. We first prove that the traffic of the link is predictable by chaos theory and introduce our traffic prediction model and the model integration. Finally, the experimental results show that our traffic prediction model could predict the traffic of link in the broker well and the strategy integrated into our traffic

prediction model could reduce the traffic cost of the overlay network efficiently.

Conflicts of Interest

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

Acknowledgments

This work was sponsored by the National Science Foundation of China, NSFC no. 61173177.

References

[1] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Computing Surveys*, vol. 35, no. 2, pp. 114–131, 2003.

[2] J. Yin, W. Lo, S. Deng, Y. Li, Z. Wu, and N. Xiong, "Colbar: a collaborative location-based regularization framework for QoS prediction," *Information Sciences*, vol. 265, pp. 68–84, 2014.

[3] M. Migliavacca and G. Cugola, "Adapting publish-subscribe routing to traffic demands," in *Proceedings of the Inaugural International Conference on Distributed Event-Based Systems (DEBS '07)*, pp. 91–96, ACM, Ontario, Canada, June 2007.

[4] Y. Yin, S. Aihua, G. Min, X. Yueshen, and W. Shuoping, "QoS prediction for web service recommendation with network location-aware neighbor selection," *International Journal of Software Engineering and Knowledge Engineering*, vol. 26, no. 4, pp. 611–632, 2016.

[5] M. Chi, S. Liu, and C. Hu, "Self-adapting routing overlay network for frequently changing application traffic in content-based publish/subscribe system," *Mathematical Problems in Engineering*, vol. 2014, Article ID 362076, 2014.

[6] M. A. Jaeger, H. Parzyjegla, G. Mühl, and K. Herrmann, "Self-organizing broker topologies for publish/subscribe systems," in *Proceedings of the ACM Symposium on Applied Computing (SAC '07)*, pp. 543–550, ACM, March 2007.

[7] H. R. Maier and G. C. Dandy, "Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications," *Environmental Modelling and Software*, vol. 15, no. 1, pp. 101–124, 2000.

[8] T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka, "Stock market prediction system with modular neural networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '90)*, pp. 1–6, June 1990.

[9] B. L. Smith and M. J. Demetsky, "Short-term traffic flow prediction: neural network approach," Transportation Research Record 1453, 1994.

[10] K. T. Alligood, T. D. Sauer, and J. A. Yorke, *Chaos*, Springer, New York, NY, USA, 1996.

[11] J. Gleick, *Chaos: Making a New Science*, Random House, 1997.

[12] F. Fernández-Rodríguez, S. Sosvilla-Rivero, and J. Andrada-Félix, "A new test for chaotic dynamics using Lyapunov exponents," Documento de Trabajo 9, 2003.

[13] M. T. Rosenstein, J. J. Collins, and C. J. De Luca, "A practical method for calculating largest Lyapunov exponents from small data sets," *Physica D: Nonlinear Phenomen*, vol. 65, no. 1-2, pp. 117–134, 1993.

[14] H. S. Kim, R. Eykholt, and J. D. Salas, "Nonlinear dynamics, delay times, and embedding windows," *Physica D: Nonlinear Phenomena*, vol. 127, no. 1-2, pp. 48–60, 1999.

[15] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[16] C. N. Lu, H.-T. Wu, and S. Vemuri, "Neural network based short term load forecasting," *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 336–342, 1993.

[17] W. Galuba, K. Aberer, Z. Despotovic, and W. Kellerer, "ProtoPeer: a P2P toolkit bridging the gap between simulation and live deployement," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Rome, Italy, March 2009.

[18] G. Mühl, L. Fiege, and P. Pietzuch, *Distributed Event-Based Systems*, vol. 1, Springer, Heidelberg, Germany, 2006.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 3824765, 12 pages
https://doi.org/10.1155/2017/3824765

Research Article

A Parallel Strategy for Convolutional Neural Network Based on Heterogeneous Cluster for Mobile Information System

Jilin Zhang,[1,2,3,4] Junfeng Xiao,[1,2] Jian Wan,[1,2,4,5] Jianhua Yang,[6] Yongjian Ren,[1,2] Huayou Si,[1,2] Li Zhou,[1,2] and Hangdi Tu[1,2]

[1]*School of Computer and Technology, Hangzhou Dianzi University, Hangzhou 310018, China*
[2]*Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, Hangzhou, China*
[3]*College of Electrical Engineering, Zhejiang University, Hangzhou 310058, China*
[4]*School of Information and Electronic Engineering, Zhejiang University of Science & Technology, Hangzhou 310023, China*
[5]*Zhejiang Provincial Engineering Center on Media Data Cloud Processing and Analysis, Hangzhou, Zhejiang, China*
[6]*College of Computer Science and Technology, Zhejiang University, Hangzhou 310018, China*

Correspondence should be addressed to Jian Wan; wanjian@hdu.edu.cn

Received 25 January 2017; Accepted 23 February 2017; Published 21 March 2017

Academic Editor: Jaegeol Yim

With the development of the mobile systems, we gain a lot of benefits and convenience by leveraging mobile devices; at the same time, the information gathered by smartphones, such as location and environment, is also valuable for business to provide more intelligent services for customers. More and more machine learning methods have been used in the field of mobile information systems to study user behavior and classify usage patterns, especially convolutional neural network. With the increasing of model training parameters and data scale, the traditional single machine training method cannot meet the requirements of time complexity in practical application scenarios. The current training framework often uses simple data parallel or model parallel method to speed up the training process, which is why heterogeneous computing resources have not been fully utilized. To solve these problems, our paper proposes a delay synchronization convolutional neural network parallel strategy, which leverages the heterogeneous system. The strategy is based on both synchronous parallel and asynchronous parallel approaches; the model training process can reduce the dependence on the heterogeneous architecture in the premise of ensuring the model convergence, so the convolution neural network framework is more adaptive to different heterogeneous system environments. The experimental results show that the proposed delay synchronization strategy can achieve at least three times the speedup compared to the traditional data parallelism.

1. Introduction

Mobile devices are involved in our daily life, from online shopping to social connection to working assistant. From the business perspective, the information gathered from the devices is so valuable that it can be used to learn customers' expense characteristics and improve user experience. For example, location is one of the key factors that are related to users' consumption behavior. We can recommend nearby restaurants, shopping malls, and parks based on location, and the function is implemented in many map applications like Google Maps. For merchants, understanding user behavior is very helpful to proactively provide potential services and increase customer engagements [1, 2]. To take advantage of the data analysis benefits, the first step is to figure out customers' consumption patterns. In this paper, we proposed a novel delay synchronization based machine learning strategy to improve pattern recognition and it laid a foundation for intelligent business marketing.

Due to the increasing data volume, the data processing is a huge challenge in mobile information systems. With the development of machine learning, convolution neural network has become a suitable method to deal with such large data. Convolutional neural network is a special multistage global training deep neural network model produced for two-dimensional image recognition [3], which combines

the traditional artificial neural network with deep learning model. It not only has the general characteristics of the traditional artificial neural network, such as nonlinear, unlimited, nonstationary, and nonconvexity characteristics [4], but also contains more advantages, including fault-tolerant ability, self-learning ability and localized receptive fields, weight sharing, and pooling (secondary sampling). Convolutional neural network can discover the characteristics directly from a large number of image data and make a more profound description of the vast amount of information contained in the image. Convolutional neural network can achieve more than two orders of magnitude improvement compared to the human identification accuracy [5, 6]. With its powerful learning ability, convolutional neural network has been widely used in target tracking [7, 8], face detection [9–11], license plate detection [12], and handwriting recognition [13, 14]. It is an important research topic in machine learning, computer vision, mobile information system, and other scientific research fields.

The previous research has shown that the low model quality can be improved by the large-scale iterative training process, through modifying, testing, and evaluating the parameters of the model (network structure, the initial value of the range, learning methods, learning rate, etc.). However, with the increasing of the model scale and the training data, the time complexity of the training process is so large [15, 16] which restricts the development of the convolutional neural network. Several researchers [15, 17–20] proposed leveraging a distributed and parallel's data processing technology to support the rapid expansion of the model scale and data size.

Existing convolutional neural network training framework often takes the algorithm characteristics and the specific machine attributes as the main basis for the system design and optimization; however, this approach does not consider the relationship of the parallel computing model and the common characteristics in machine learning applications. For example, in the system architecture, (1) multicore and many-core technology has been widely used in parallel computers to increase the processing speed; besides, in order to reduce the cost, heterogeneous cluster has gradually replaced the traditional custom machine and becomes the mainstream architecture structure. So the traditional parallel computing is not suitable for the new era of big data parallel computing system; (2) the traditional parallel computing mode is of vertical expansion by leveraging more computing resources to enhance the performance, but terabyte or petabyte data processing and analysis require the horizontal expansion to improve the performance. Due to the different expansion requirements, the traditional methods of parallel computing are difficult to solve modern big data application issues.

In the general characteristics of convolution neural network application, the characteristics of parallel applications changed. Machine learning or deep learning application is a typical intensive computing iterative convergence application which shows the characteristics of fault-tolerance, structural dependence, nonuniform convergence, and sparse optimization. Traditional parallel computing application guarantees the accuracy of data parallelism or model parallelism by large amount of data synchronization; it cannot make full use of the characteristics of dense computing iterative convergence algorithm to improve the performance of application.

The basic idea of existing large-scale convolutional neural network parallelization is through reconstruction of convolutional neural network algorithm to give full play to the system performance advantages, in order to improve training efficiency and effect. However, such a program usually has two key issues. First, optimization methods issue means how to choose the optimization method to improve the efficiency of intensive computing iterative convergence algorithm. Second, the allocation of machine resources and data communication between nodes all require developers to manually perform single static tuning. Not only is it for too long, but it also relies on the experience of developers heavily. It is difficult to adapt to the structural changes in computing resources.

In this paper, we proposed a new delay synchronization parallel strategy for heterogeneous distributed cluster system. It is based on the traditional data parallel and model parallel method and combined with stale synchronous parallel parameter server system [21]. This strategy can shield the factors in model training process such as the communication bandwidth, memory bandwidth, memory hierarchy, memory latency, thread management, and processing mode. The training process will not be affected by the dynamic changes of the computing resources if the resources are adequate. As a result of decoupling the training algorithm and system hardware resources, the proposed strategy successfully frees developers from process calculation, resource allocation, and data communication optimization, and it effectively improves the program, especially in the heterogeneous environments.

Section 2 introduces an overview of the convolutional neural network parallel strategy. Section 3 gives the problem description, including training methods and existing problems of data parallel and model parallel. Section 4 describes the process of delay synchronous parallel strategy. Section 5 presents the experimental results and corresponding analysis. Section 6 summarizes the paper.

2. Related Work

This section mainly describes the strategies related to convolution neural network parallelization, including several early methods of parallelization and the current mainstream distributed data parallelization and model parallelization method.

2.1. Early Parallelization Methods. FPGA (field programable gate array), as a kind of computing intensive accelerator, can accelerate algorithm by mapping it to the hardware module. We usually use the method that combined the "host" with "FPGA" [22], and the host used in the control of training process of beginning and ending provides image data as input in the forward propagation. The application of FPGA based artificial neural network includes image segmentation [23], image and video processing [24], intelligent image analysis [25, 26], autonomous robot technology [27], and sensorless control [28, 29]. But because this kind of parallel method requires the programer to have the solid digital

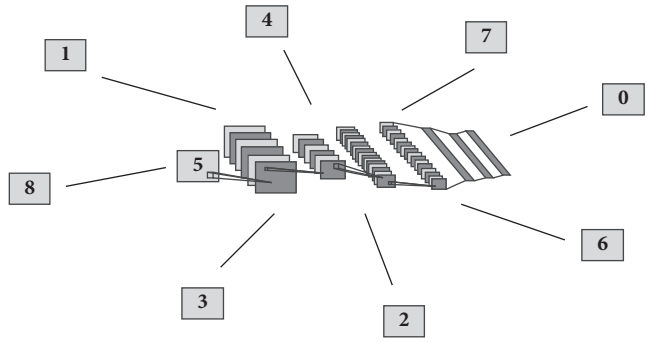


Figure 1: The schema of data parallel.

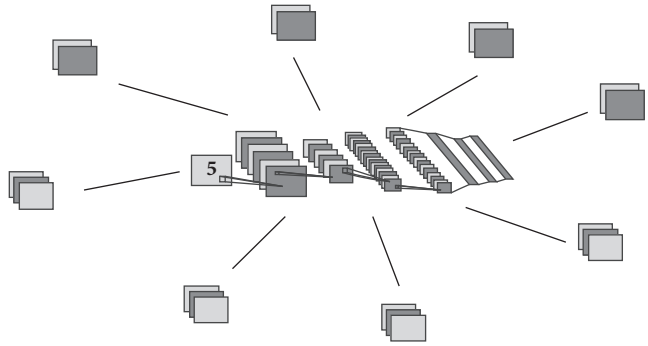


Figure 2: The schema of model parallel.

circuit knowledge and the programing complexity is high, it is barely used in practice.

With the rapid development of the GPU, the floating point operation speed is 10 times faster than the same period CPU; the researchers began to use GPU to accelerate the convolution neural network algorithm [30–32]. GPU contains a large (up to 10 G) shared memory and thousands of streaming processors, suitable for the inherent parallel structure of the convolutional neural network. GPU indeed can accelerate the performance [33, 34], but, taking into account the heterogeneous system, mapping algorithm to the hardware system cannot give full play to the resources of computing hardware.

When faced with the massive data of terabyte or petabyte, MapReduce is used to solve the problem. Convolutional neural network based on MapReduce parallel [35, 36] can also achieve a better result, but with the increasing of the number of parameters in the network model, and the difficulty of model training are increased as well. MapReduce is not suitable for high computing density iterative algorithm.

2.2. Data Parallel and Model Parallel. Data parallel and model parallel were proposed by Google distributed researcher Jeff Dean and deep learning researcher Andrew Ng in 2012 on the project called "Google Brain" [37, 38]. It referred to use of CPU cluster architecture combined with model parallel and data parallel implementation of the deep learning system DistBelief.

Data parallel means that, in the process of the model training, the training samples are divided and distributed to different computing nodes, and each computing node has a training model. After the end of each iteration, each node is doing a weight update communication to update the training model. Data parallel schema is shown in Figure 1 [37, 38].

In the model parallel, the model is divided into multiple slices, each of which is stored into a single server, all of which can be trained for a complete model. In the process of the training, each node contains a complete model network but only trained a specific part of the model. The model parallel is more suitable for the large model; it can solve the problem of limited training memory on a single machine. Adopting the model parallel distributed method can reduce the size of the occupied memory in each node. The model parallel scheme is shown in Figure 2 [37, 38].

By using the two methods in heterogeneous system, it is difficult for the convolutional neural network algorithm to select the appropriate optimization method and the optimal time, which means it cannot give full play to the advantages of the computing resources of the heterogeneous system. Moreover, when the hardware condition changes, the training algorithm cannot dynamically adapt to the computing resources, so the efficiency of the training process is not high.

In short, the existing parallel methods cannot fully adapt to the heterogeneous architecture, so as not to take advantages of heterogeneous architecture resource.

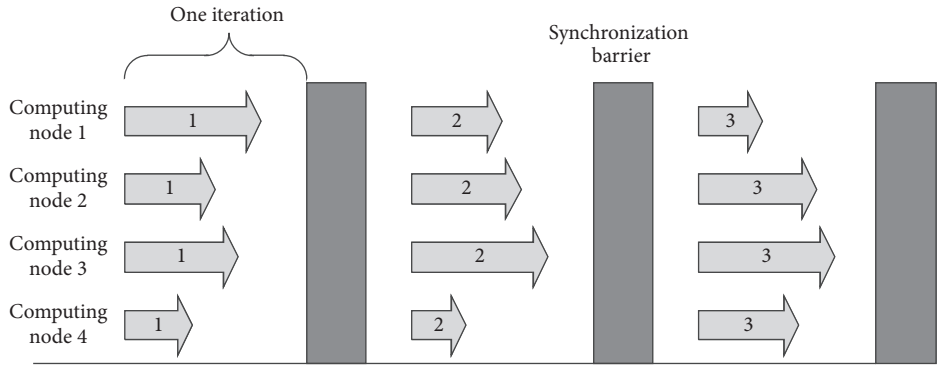


Figure 3: Training process of synchronous parallel.

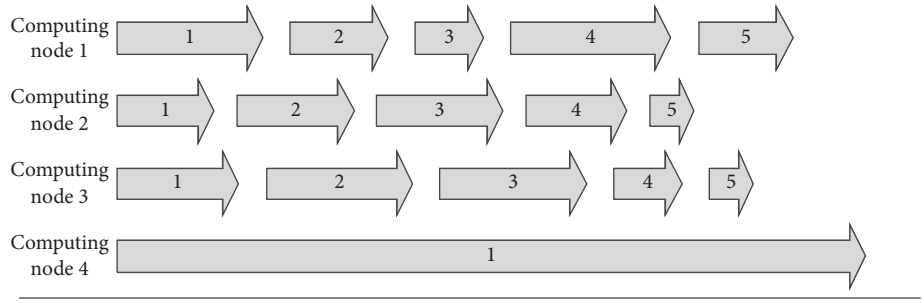


Figure 4: Training process of asynchronous parallel.

3. Problem Description

In this paper, we mainly introduce how to train the neural network in the heterogeneous and distributed environment. In this section, we first describe the training methods and existing problems of data parallel and model parallel and then further explain the factors that should be considered in the parallelization process.

3.1. Data Parallel and Model Parallel Training Methods. Both data parallel and model parallel can be categorized as synchronous parallel and asynchronous parallel, parameter server parallel and nonparametric server parallel.

Synchronous and asynchronous parallel processes are shown in Figures 3 and 4. Synchronous parallel method means that, in the process of model training, each update of the training model is carried out after the completion of an iteration of all computing nodes. And each node begins to continue the next iteration of the training after they obtain a new training model. But, in the method of asynchronous parallel, when the iteration is completed, faster computing node notices other computing nodes to update the weights after the completion of one iteration but does not wait for other nodes to be updated.

From Figures 3 and 4, we can see that the synchronous parallel calculation requires the use of synchronization barrier to force all the computing nodes to carry out a parameter update after one iteration is completed. This parallel method will lead to faster nodes waiting for the other slower nodes, which greatly affected the model training speed. So in the synchronous parallel training method, in order to obtain a better training speed, the load balance between each computing node is a stringent requirement. In practical training, the current performance of the node is affected by the external environment and other tasks in the computing nodes. Hence, the current performance of the node is random, eventually leading to the performance of the synchronous parallel method being dragged down by the slowest computing nodes. Because the model update is completed at the same time, it will take up a lot of memory and generate a data communication storm, making a higher requirement for computing nodes.

In asynchronous parallel computing method, each node directly updated model parameters immediately after the calculation completion without waiting for other nodes to complete their iteration. Asynchronous parallel method does not need to consider the performance of computing nodes; it only needs to focus on the calculation of the node itself. Asynchronous parallel method in N nodes can get almost N times the speedup. But, in the asynchronous parallel mode, the training of the model parameters is not the newest, making the training process easy to fall into local optimal solution, resulting in poor network training convergence. So the asynchronous parallel method cannot be used to model training in practice.

As Figure 5 shows, in the process of model training, the parameter server is used to complete the update process after each iteration. Parameter servers can also be responsible for

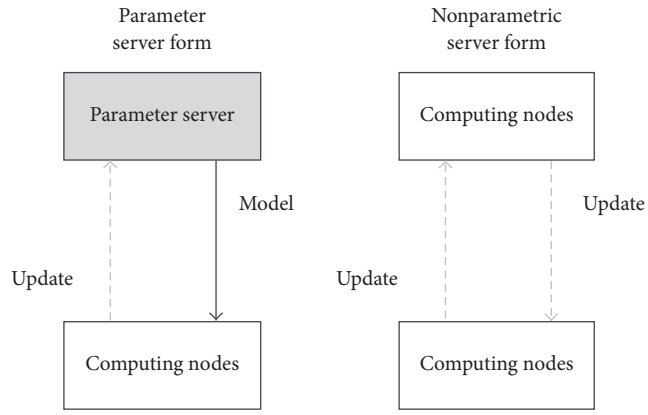


Figure 5: Comparison between parameter server parallel and nonparametric server parallel.

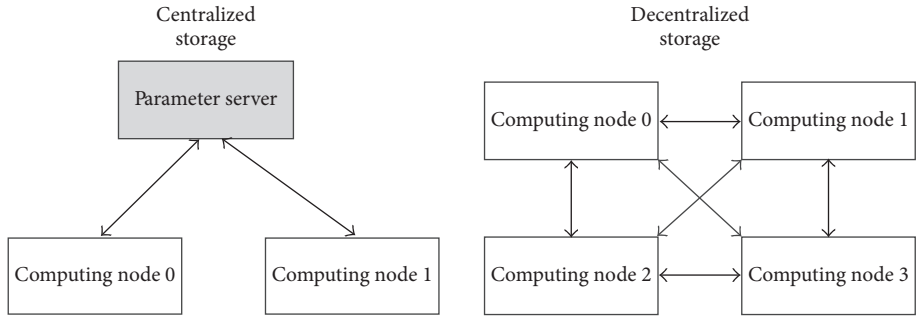


Figure 6: Communication topology comparison between centralized storage and decentralized storage.

sending training data and test data. When the parameter server is used, each computing node does not communicate with the other; they communicate with the parameter server.

3.2. Considerations for Parallel. From the previous discussion of data parallel and the model parallel training methods, we summarize some factors which should be considered in the implementation of the heterogeneous system.

3.2.1. Parallel Method. In the heterogeneous system, data parallel was chosen to train the model in general. The reasons are as follows: (1) in data parallel, each node has the same training method for the network model. But, in the model parallel, each node does not have the same training method for the network model (because each node is training different part of the model). So it is more difficult to implement the model parallel compared to the data parallel. (2) Convolutional neural network has the structure dependence; the model parameter matrix update order will affect the time of model training. Moreover, because the network has fault-tolerant ability, it will recover from error which was caused by the unreasonable task division. When the error accumulated to a certain degree, the network model may get a local optimal solution. In addition, the division of the model is lack of theoretical guidance.

3.2.2. Maximizing Effective Training Time: Delay Synchronous Parallel. In the distributed and synchronous parallel environment, the calculation nodes have to wait to synchronize the parameters after each iteration in the training process. In order to reduce the waiting time of computing nodes, the load balancing between each node is required. However, in the actual situation, the performance of the machine is often affected by a lot of external factors, such as temperature, and they is random factors. In the case of asynchronous parallel environment, the computing of each node does not interfere with that of the other. The faster nodes do not have to wait for the slower nodes, and they can directly update network model. This training mode is equivalent to shielding the impact of different hardware computing capabilities. Based on the characteristics of synchronous parallel and asynchronous parallel, the delay synchronous parallel is proposed. See Section 4 for details.

3.2.3. Parameter Storage and Communication Topology. The choice of parameter storage will affect the communication topology, and the topology of the communication will influence the weight parameters communication between each node. Depending on whether or not the parameter server is used, the storage of the parameters can be divided into centralized storage and decentralized storage. The communication topology is shown in Figure 6. Centralized storage can use the "master and slave" mode for implementation, the

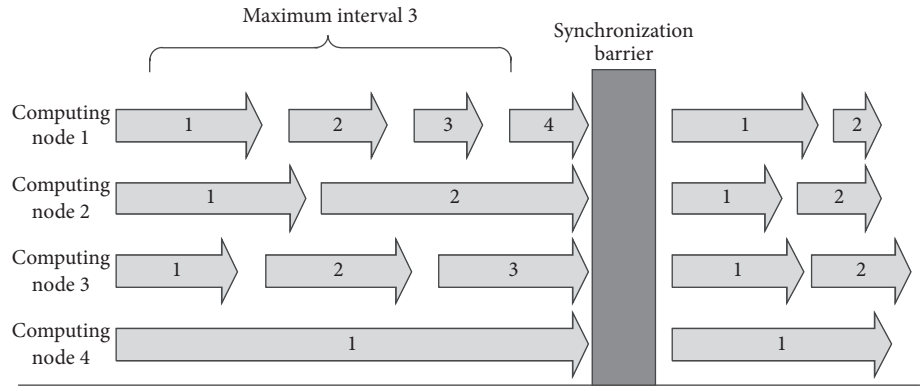


Figure 7: Training method of delay synchronization parallel.

master node acts as a parameter server and a data transmitter, and the slave node is used for training the model. Parameter server contains a complete model and sends the model to each slave node before training. The weight update will be sent to the parameter server after the slave completes one iteration. Parameter server will update the network model as soon as it gets all the weight updates from slave nodes. Then the master node will send the latest network model to slave node for training. Because the centralized storage only needed to send the weight update data to the master node from the slave node, there is no exchange of data between slave nodes, which greatly reduces the cost of communication. Decentralized storage can usually be implemented by the end-to-end topology, and each endpoint is a calculation node. Each node will send the weight update data to others after one iteration is complete; this will cause too much communication during the whole process of distributed training. In order to reduce the communication overhead in the training process, the centralized storage is used to implement the distributed training.

In the specific implementation, centralized storage is more challenging than decentralized storage. Firstly, the master node in the centralized storage has high performance requirements to coordinate the whole training process. Secondly, concurrency control between nodes should be considered in centralized storage. Finally, we need to consider the storage mode of the training data to reduce communication loss.

4. Delay Synchronization Parallel

From the section entitled "Maximizing Effective Training Time: Delay Synchronous Parallel," we know the purpose of the delay synchronization parallel method is to ensure that the model is not trapped in local optimal solution, and the effective training time of the nodes is maximized. The synchronous parallelism can ensure that the training process does not fall into local optimal solution, and the asynchronous approach can make the effective training time maximized. So the delay synchronization parallel approach combines the advantages of synchronous and asynchronous parallelism. In this section, we first describe the training method of the delay synchronization, then introduce the training characteristics of this method, and finally show the conditions the node should have to achieve for this method.

4.1. Training Method. For a network model training, assuming that there are P calculation nodes, after the end of each iteration, training faster computing nodes need to wait for the slower training nodes to finish in synchronous update method. While using delay synchronization in the parallel way, faster computing node does not need to stop to wait for the slower computing nodes, and faster computing node can directly update the network model parameters and then continue to the next iterative training. As Figure 7 shows, when the slowest computing node is slower by s (s value can be set by the user) times of iterations than the fastest node, the fastest node is forced to wait until all the computing nodes complete their one iteration, and then a training model parameter update between all computing nodes is completed.

4.2. Training Characteristics. In order to reduce the influence of communication process, we apply the server parameter to implement delay synchronous parallel method; it means parameters are stored and updated by the centralized node and all the computing nodes only need communication with the server parameter. Moreover, the update of the model also depends on the parameters server for completion.

Synchronous parallel computing requires the use of the synchronization barrier to force all the computing nodes to do a parameter update after completing an iteration. In the training method of synchronous parallel, in order to get a better training speed, the load balance between the nodes is strict. But, in the asynchronous parallel, each computing node does not wait for the others after they complete one iteration, and the completed nodes will directly update their training model through parameter server. Asynchronous parallel method does not need to consider the performance of computing node; it only needs to focus on the calculation of the computing nodes. It can be said that the asynchronous parallel way can shield a series of problems caused by the uneven performance between calculation nodes.

Delay synchronization parallel contains the characteristics of synchronous parallel and asynchronous parallel;

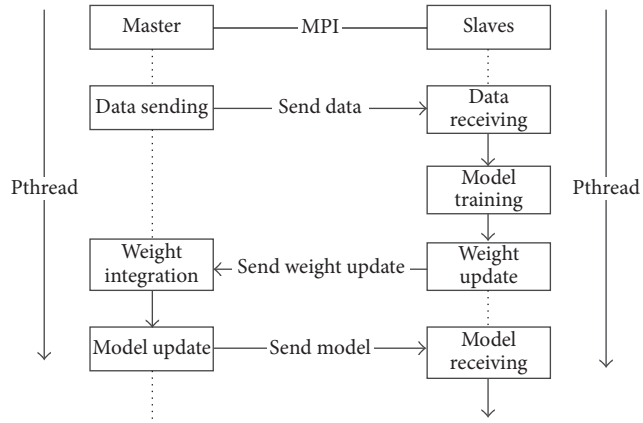


Figure 8: Training process of "master and slave" model.

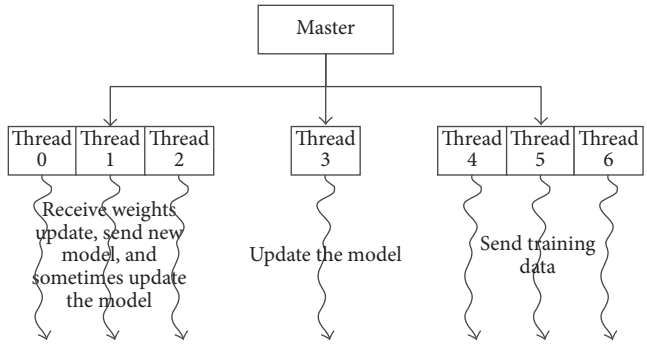


Figure 9: Design of the main process.

we adopt the asynchronous training approach in the early training phase, when the difference between the fastest and the slowest nodes is s iterations, using the synchronous barrier to mandate all nodes to do a model update. All nodes will continue to do asynchronous training until next s iteration emerged. In the process of asynchronous training, the training method can shield the effect caused by different node performance, and the synchronous barrier can avoid the local optimal solution. Therefore, the parallel method of delay synchronization can get the same speedup as the computation node as well as a better training result.

4.3. Implementation Conditions. In order to implement the delay synchronization, the algorithm must meet the following conditions: (1) the fastest node and the slowest node work even if the interval is less than the number of s. (2) Each computing node has a training model, with noninterference between others. (3) Third one is using a parameter server to update the model parameters and undertake date distribution function.

5. Experimental Results and Discussions

In this section, we verify the effectiveness, performance, and scalability of the delay synchronization parallel strategy and present the influence of different maximum interval s on the training results. The data set we used is the MNIST handwritten digital font data set which includes 60000 pictures' training data and 10000 pictures' test data. We use the classic LeNet-5 model for training, and the model includes one input layer, one output layer, three convolutional layers, two pooling layers and a fully connected layer. The batch size of the training model is 64, and the maximum number of iterations is 10000.

5.1. Experimental Framework. The training environment is based on the MPI master-slave model of distributed data parallel, the specific training process and the detailed design of the master and slave nodes are shown in Figures 8, 9, and 10, respectively.

The main process consists of three thread groups: the data distribution thread group, the parameter communication thread group, and the model update thread. Data distribution thread group and parameters communication thread group have the same thread number which is the number of computing processes. The data distribution thread group is mainly used for distributing the training task and data to each computing process. The parameters communication thread group is mainly used for receiving the weight update data sent by the computing processes and sending the new model for computing processes after the model is updated. When the interval between the fastest node and the slowest node

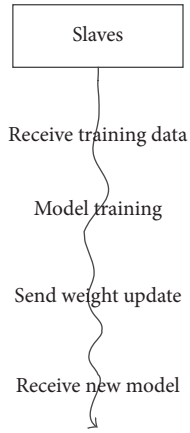


Figure 10: Design of the computing process.

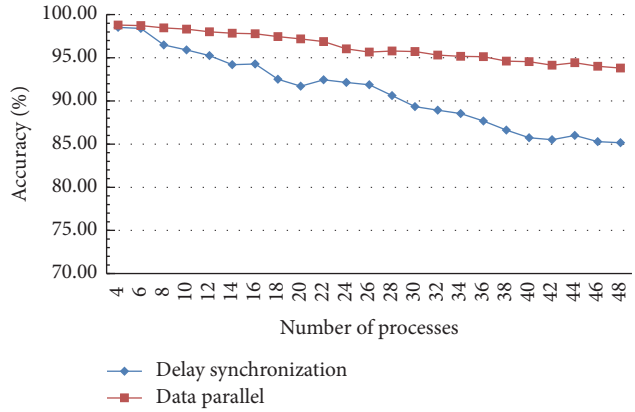


Figure 11: Delay synchronization parallel performance test: accuracy.

is less than the maximum interval number s, parameters communication thread group also used to update the model. Model is trained by the computing processes in a serialization manner.

5.2. Effectiveness and Performance. In this section, we will verify the effectiveness of the delay synchronization strategy and evaluate its performance through experiments. Hardware environment used in the experiments consists of two heterogeneous server nodes connected by Gigabit Ethernet. One node is configured with the 24 Intel Xeon E5-2620 V2 @2.10 GHz CPU and 128 G memory, and the operating system is Red Hat Enterprise Linux Server release 6.3. The other node is configured with the 32 Intel Xeon E5-2670 @2.6 GHz CPU and 32 G memory, and the operating system is Red Hat Enterprise Linux Server release 6.2. We compare the performance of traditional data parallel and delay synchronous parallel from aspects of time and accuracy. The specific experimental results are shown in Figures 11 and 12, where the maximum interval s is set to 3, and the recording time includes all the time from the distribution of the model to the time of testing test data.

From Figures 11 and 12, we can see that, in the traditional data parallel and delay synchronous parallel training methods, the accuracy rate is decreased with the increase of computing nodes, and delay synchronization parallel accuracy rate declines faster than traditional data parallel. Because of the communication cost, the training time is nonlinearly reduced with the increasing of the computing processes. When passing a certain computing process number, the time even increased. In the best case, the delay synchronization parallel strategy can get almost three times faster than the traditional data parallel. When the process number is 10 and we add other unrelated process tasks in the training environment, the training time of data parallel method increased, but the delay synchronization scheme was not affected. It can be seen that the delay synchronization parallel strategy reduces the impact of the hardware environment; that is, the training time is not easy to be dragged by a short board of computing process.

5.3. Scalability. In order to verify the scalability of the delay synchronization parallel strategy, the experiment environment consists of four heterogeneous servers connected with the Gigabit Ethernet. One node is configured with the 24 Intel Xeon E5-2620 V2 @2.10 GHz CPU and 128 G memory, and the operating system is Red Hat Enterprise Linux Server release 6.3. Another three nodes are configured with the 32

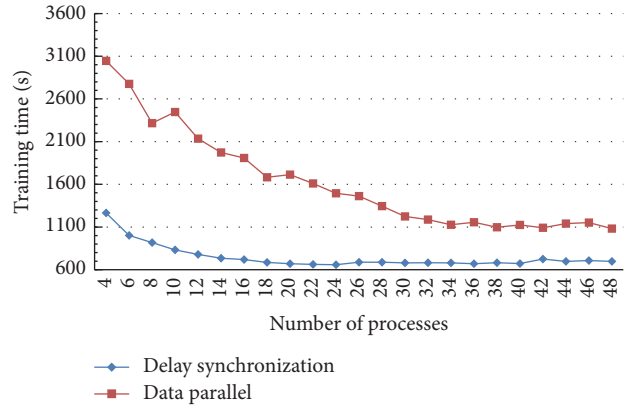


Figure 12: Delay synchronization parallel performance test: training time.

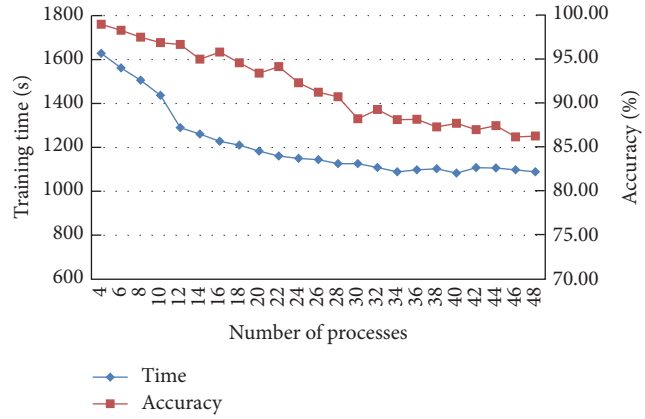


Figure 13: Verification of scalability.

Intel Xeon E5-2670 @2.6 GHz CPU and 32 G memory, and the operating system is Red Hat Enterprise Linux Server release 6.2. The evaluation metrics of the experiment are the same as the section entitled "Effectiveness and Performance," which are time and accuracy. In order to balance the iterative tasks on each node, the training processes are distributed on the four nodes, the maximum number of intervals in delay synchronization parallel training is still three, and the specific experimental results are in Figure 13.

From Figure 13, for accuracy, the training of four nodes has almost the same effect as described in the section entitled "Effectiveness and Performance," and it presents a downward trend with the increment of computing processes. From the perspective of training time, compared to experiments in section entitled "Effectiveness and Performance," the overall training time is increased as a result of more communication overhead. Same as section entitled "Effectiveness and Performance," the training time is nonlinearly decreased with the increasing of the computing processes. Time may increase after the number of computing processes passes a certain value. The experimental results show that the delay synchronization strategy has good scalability, but this kind of good scalability is inevitable to involve a certain amount of communication cost.

5.4. The Maximum Interval Influence on Model Training Process. Delay synchronous parallel strategy is a combination of the synchronous parallel strategy and asynchronous parallel strategy. When the fastest node is s (the maximum interval) iteration(s) faster than the slowest node, the strategy uses the mandatory synchronization barrier to prevent the model divergence from falling into local optimal solution. This section verifies the effects of different maximum interval s on the model training. The experimental environment is the same as described in section entitled "Effectiveness and Performance," which is the two heterogeneous servers connected by Gigabit Ethernet. The time and the effect of the training process were tested with the maximum interval of 1, 2, and 3, and the results are presented in Figures 14 and 15.

From Figures 14 and 15, we can see that, with the increasing of the maximum interval, model training time decreases, but the accuracy of the model was effected dramatically with the computing process increase. Hence, considering the influence of both time and accuracy, we prefer to select smaller interval.

Based on the experimental results, we can see that the proposed delay synchronization parallel strategy indeed has a better performance.

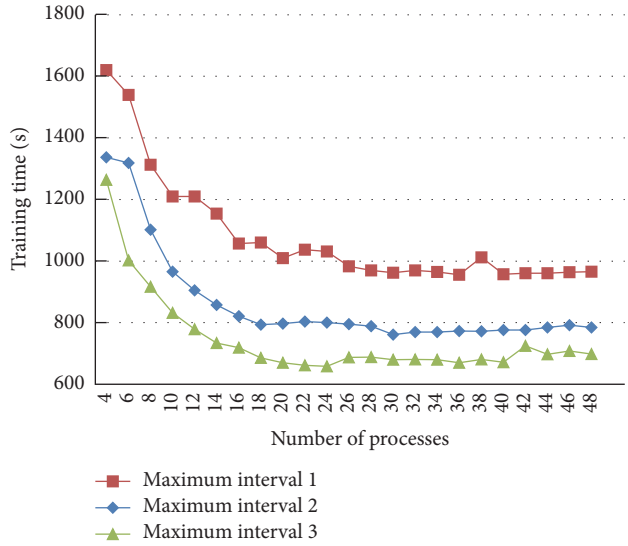


Figure 14: The maximum interval influence on model training process: training time.

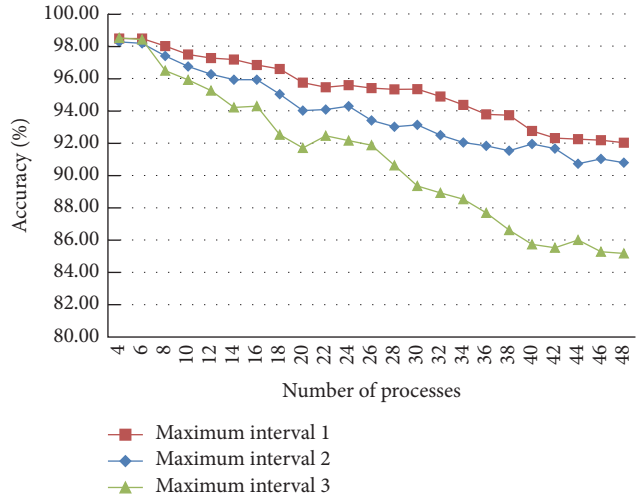


Figure 15: The maximum interval influence on model training process: accuracy.

6. Conclusion

Mobile device is an integral part of our daily life; business market can be more intelligent to automatically provide services based on users' location and context environment. To learn users' habits and patterns, machine learning strategies, such as CNN, are applied. However, the existing parallel implementation cannot fully use the parallel computing architecture resources, making heterogeneous computing resources wasted, especially in the mobile information system field. To this end, this paper proposes a convolutional neural network parallel strategy based on the heterogeneous clusters named delay synchronization parallel strategy. The strategy leverages the benefits of both synchronous parallel and asynchronous parallel approaches. It can achieve almost 3 times the speedup compared to the data parallel. The scalability of the strategy can make convolution neural network framework more adaptive to different heterogeneous system environments.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Jilin Zhang and Junfeng Xiao contributed equally to this work and should be considered co-first authors.

Acknowledgments

This work is partly supported by the National Natural Science Foundation of China under Grants no. 61672200

and no. 61572163; the National High Technology Research and Development Program of China under Grant no. 2015AA01A303; the Zhejiang Natural Science Funds under Grants no. LY16F020018 and no. LY17F020029; the Key Laboratory of Complex Systems Modeling and Simulation program of the Ministry of Education and the Chinese Postdoctoral Science Foundation no. 2013M541780 and no. 2013M540492; Hangzhou Dianzi University construction project of graduate enterprise innovation practice base no. SJJD2014005; Research project of Zhejiang Provincial Department of Education under Grant no. Y201016492.

References

[1] P. Racherla, C. Furner, and J. Babb, "Conceptualizing the implications of mobile app usage and stickiness: a research agenda," 2012.

[2] M. Gençer, G. Bilgin, Ö. Zan, and T. Voyvodaoglu, "A new framework for increasing user engagement in mobile applications using machine learning techniques," in *Proceedings of the International Conference on Design, User Experience, and Usability*, pp. 651–659, Springer, Las Vegas, Nev, USA, 2013.

[3] B. B. Le Cun, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems*, pp. 396–404, 1990.

[4] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.

[5] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 3642–3649, Providence, RI, USA, June 2012.

[6] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Networks*, vol. 32, pp. 333–338, 2012.

[7] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1610–1623, 2010.

[8] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2011.

[9] S. Naji, R. Zainuddin, S. A. Kareem, and H. A. Jalab, "Detecting faces in colored images using multi-skin color models and neural network with texture analysis," *Malaysian Journal of Computer Science*, vol. 26, no. 2, pp. 101–123, 2013.

[10] N. Rajput, P. Jain, and S. Shrivastava, "Face detection using HMM-SVM method," in *Advances in Computer Science, Engineering & Applications*, pp. 835–842, Springer, Berlin, Germany, 2012.

[11] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 3476–3483, IEEE, Portland, Ore, USA, June 2013.

[12] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. Von Deneen, and P. Shi, "An algorithm for license plate recognition applied to intelligent transportation system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 830–845, 2011.

[13] D. V. Phạm, "Online handwriting recognition using multi convolution neural networks," in *Proceedings of the Asia-Pacific Conference on Simulated Evolution and Learning*, pp. 310–319, Springer, Hanoi, Vietnam, 2012.

[14] S. S. Ahranjany, F. Razzazi, and M. H. Ghassemian, "A very high accuracy handwritten character recognition system for Farsi/Arabic digits using convolutional neural networks," in *Proceedings of the IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA '10)*, pp. 1585–1592, IEEE, Changsha, China, September 2010.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, December 2012.

[16] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15)*, pp. 1–9, Boston, Mass, USA, June 2015.

[17] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," https://arxiv.org/abs/1404.5997.

[18] Y. Jia, E. Shelhamer, J. Donahue et al., "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM Conference on Multimedia (MM '14)*, pp. 675–678, November 2014.

[19] J. Yin, X. Lu, C. Pu, Z. Wu, and H. Chen, "JTangCSB: a cloud service bus for cloud and enterprise application integration," *IEEE Internet Computing*, vol. 19, no. 1, pp. 35–43, 2015.

[20] D. Povey, A. Ghoshal, G. Boulianne et al., "The Kaldi speech recognition toolkit," in *Proceedings of the IEEE 2011 Workshop on Automatic Speech Recognition and Understanding (No. EPFL-CONF-192584)*, IEEE Signal Processing Society, Waikoloa, Hawaii, USA, 2011.

[21] Q. Ho, J. Cipar, H. Cui et al., "More effective distributed ML via a stale synchronous parallel parameter server," in *Advances in Neural Information Processing Systems*, pp. 1223–1231, 2013.

[22] C. Farabet, B. Martini, P. Akselrod, S. Talay, Y. LeCun, and E. Culurciello, "Hardware accelerated convolutional neural networks for synthetic vision systems," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '10)*, pp. 257–260, IEEE, Paris, France, May-June 2010.

[23] R. Hemalatha, N. Santhiyakumari, and S. Suresh, "Implementation of medical image segmentation using Virtex FPGA kit," in *Proceedings of the 4th International Conference on Signal Processing and Communication Engineering Systems (SPACES '15)*, pp. 358–362, January 2015.

[24] J. G. Pandey, A. Karmakar, and S. Gurunarayanan, "Architectures and algorithms for image and video processing using FPGA-based platform," in *Proceedings of the 18th International Symposium on VLSI Design and Test (VDAT '14)*, July 2014.

[25] E. C. Pedrino, O. Morandin Jr., E. R. R. Kato, and V. O. Roda, "Intelligent FPGA based system for shape recognition," in *Proceedings of the 7th Southern Conference on Programmable Logic (SPL '11)*, pp. 197–202, IEEE, Córdoba, Spain, April 2011.

[26] A. Zawadzki and M. Gorgoń, "Automatically controlled pan-tilt smart camera with FPGA based image analysis system dedicated to real-time tracking of a moving object," *Journal of Systems Architecture*, vol. 61, no. 10, pp. 681–692, 2015.

[27] T. Nakamura, Y. Touma, H. Hagiwara, K. Asami, and M. Komori, "Scene recognition based on gradient feature for autonomous mobile robot and its FPGA implementation," in *Proceedings of the 4th International Conference on Informatics,*

Electronics and Vision (ICIEV '15), pp. 1–4, IEEE Computer Society, Kitakyushu, Japan, June 2015.

[28] L. Idkhajine, E. Monmasson, and A. Maalouf, "Fully FPGA-based sensorless control for synchronous AC drive using an extended Kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 10, pp. 3908–3918, 2012.

[29] S. Narjess, T. Ramzi, and M. M. Faouzi, "Implementation of sensorless control of an induction motor on FPGA using Xilinx system generator," *Journal of Theoretical & Applied Information Technology*, vol. 92, no. 2, pp. 322–334, 2016.

[30] K. Yu, "Large-scale deep learning at Baidu," in *Proceedings of the the 22nd ACM International Conference*, pp. 2211–2212, ACM, San Francisco, Calif, USA, October 2013.

[31] A. Coates, "Deep learning with COTS HPC systems," in *Proceedings of the International Conference on Machine Learning (ICML '13)*, pp. 1337–1345, Atlanta, Ga, USA, 2013.

[32] O. Yadan, K. Adams, Y. Taigman, and M. A. Ranzato, "Multi-GPU training of convnets," https://arxiv.org/abs/1312.5853.

[33] R. Uetz and S. Behnke, "Large-scale object recognition with CUDA-accelerated hierarchical neural networks," in *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS '09)*, vol. 1, pp. 536–541, IEEE, Shanghai, China, November 2009.

[34] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI '11)*, pp. 1237–1242, July 2011.

[35] Z. Liu, H. Li, and G. Miao, "MapReduce-based backpropagation neural network over large scale mobile data," in *Proceedings of the 6th International Conference on Natural Computation (ICNC '10)*, vol. 4, pp. 1726–1730, IEEE, Yantai, China, August 2010.

[36] Q. Wang, J. Zhao, D. Gong, Y. Shen, M. Li, and Y. Lei, "Parallelizing convolutional neural networks for action event recognition in surveillance videos," *International Journal of Parallel Programming*, 2016.

[37] Q. V. Le, "Building high-level features using large scale unsupervised learning," in *Proceedings of the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '13)*, pp. 8595–8598, Vancouver, Canada, May 2013.

[38] J. Dean, G. Corrado, R. Monga et al., "Large scale distributed deep networks," in *Advances in Neural Information Processing Systems*, pp. 1223–1231, 2012.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 6310827, 10 pages
https://doi.org/10.1155/2017/6310827

Research Article

Exploring Intracity Taxi Mobility during the Holidays for Location-Based Marketing

Wen-jun Wang, Xiao-ming Li, Peng-fei Jiao, Guang-quan Xu, Ning Yuan, and Wei Yu

School of Computer Science and Technology, Tianjin University, Tianjin 300072, China

Correspondence should be addressed to Guang-quan Xu; losin@tju.edu.cn

Received 24 January 2017; Accepted 1 March 2017; Published 20 March 2017

Academic Editor: Jaegeol Yim

Taxi mobility information can be considered as an important source of mobile location-based information for making marketing decisions. So, studying the behavioral patterns of taxis in a Chinese city during the holidays using the global positioning system (GPS) can yield remarkable insights into people's holiday travel patterns, as well as the odd-even day vehicle prohibition system. This paper studies the behavioral patterns of taxis during specific holidays in terms of pick-up and drop-off locations, travel distance, mobile step length, travel direction, and radius of gyration on the basis of GPS data. Our results support the idea of a polycentric city. It is concluded from the reporting results that there are no significant changes in the distribution of pick-up and drop-off locations, travel distance, or travel direction during holidays in comparison to work days. The results suggest that human travel by taxi has a stable regularity. However, the radius of gyration of movement by most of the taxis becomes significantly larger during holidays that indicate more long-distance travels. The current study will be helpful for location-based marketing during the holidays.

1. Introduction

Human behavior is the source of all of social phenomena, including location-based marketing. The current research is mainly focused on analyzing the human behavior quantitatively in statistical physics and complexity science. In the state of the art in the field, many researchers analyzed the human behavior in different aspects. Barabási (2005) studied the power-law characteristics in the distribution of the interevent time of human communication behavior [1]. Brockmann et al. (2006) analyzed the data of dollar bills in circulation and got the moving step of each dollar bill in space. The authors concluded that its moving step probability has obvious characteristics of power-law distribution, with an exponential power of −1.59 [2]. Similarly, Gonzalez et al. (2008) used mobile phone data to analyze the spatial distribution characteristics of moving steps by mobile phone users. They observed that the moving step is in line with a power-law distribution with an exponential power of -1.75 ± 0.15 [3]. Song et al. (2010) also reported that accuracy of predicting human behavior lies in the range of 70% to 93% [4] The results were reported on the basis of analyzing the mobile phone records of one million users for a period of three months.

Recently, it has been observed that the analysis and prediction of human spatial movement [5] is an emerging research topic [6–10] in the field relevant to urban planning [11], the spread of infectious diseases [12], and catastrophic emergency management [13] and many more. Most of the researchers in the field have only been able to analyze the behavioral patterns of human movement based upon the data collected through surveys. Nowadays, the researchers are also exploring the utilization of personal mobility data for analyzing the human behavior, such as vehicle global positioning system (GPS) data [14] and mobile phone records [3].

GPS data provides a precise spatial resolution. Its capability to represent people's mobility features has made its wide use for the analysis of human behavior. Rhee et al. (2011) collected and analyzed GPS data of 44 volunteers. Their analysis results confirm that the moving step of different groups of volunteers for different scenarios approximated a power-law distribution [15], whereas the finding of the few studies suggests an exponential distribution of distance

traveled by taxi passengers [16, 17]. There exist a significant research for the collection of the car GPS data in Rome, Bologna, Senigallia, and Florence and a lot of statistical research on private car drivers' travel trajectory. It was found that vehicle travel distance had an exponential distribution that remains invariant with the time [18–20].

In the present paper, we analyze the behavior of human spatial movement by using taxi GPS data collected over Tianjin, China. We studied the impact of holidays on human movement in various aspects, such as the distribution of urban residents' pick-up and drop-off locations by taxi, travel distances, travel directions, and taxis' scope of activities. The findings of the study are as follows:

(i) Pick-up and drop-off locations for the urban residents by taxis are mainly concentrated in the three time periods:

(a) Morning: 8:00–12:00
(b) Afternoon: 14:00–20:00
(c) Evening: 22:00–0:00 (next day).

It is observed that these locations are mainly focused between 8:00 and 10:00 in the morning during weekdays and between 10:00 and 12:00 in the morning during the holidays.

(ii) Pick-up and drop-off locations for the urban residents by taxis are mainly distributed throughout Tianjin's main urban area and the Binhai New Area, as well as two isolated hub locations: Tianjin Binhai International Airport and Tianjin South Railway Station.

A heterogeneous distribution of residents' travel distance by taxis and a centrally symmetric pattern distribution of travel direction have been observed.

The holidays do not have a significant impact on the distribution of pick-up and drop-off locations, travel distance, or travel direction.

(iii) The taxis' radius of gyration becomes significantly larger during the holidays.

The remainder of this paper is organized as follows: Section 2 describes the GPS data and its preprocessing. Section 3 presents the statistical analysis and the results. Finally, Section 4 presents the discussions of the results and concludes the paper.

2. Data Description

In the current study, we collected the GPS data of 3051 taxis in Tianjin during the month of October 2012. The important features of the collected data involve the taxi's vehicle identification number, vehicle meter status, longitude, latitude, date, and time. The sampling frequency of empty taxis is once per 20 seconds, and the sampling frequency of carrying-state taxis is once per minute. The collected data is presented in the form of records as described below:

Taxi ID: the unique ID of each taxi

Time: the sample timestamp YYYY-MM-DD HH:MM:SS

GPS position: the longitude and latitude of the sample taxi at the sample time

Meter state: indicating whether the taxi meter is running: 0 represents that there are no passengers in this taxi, and 1 represents that there are passengers in this taxi

The collected data is preprocessed for further analysis by extracting the trips of each taxi and dropping the trips beyond the scope of the city under the study.

The meter state identifies the presence of passenger(s) in the taxi. Therefore, a taxi's travel trajectory of meter state is similar to 0000001111111100000. We also extracted the O location (origin location, also called pick-up location) and D location (destination location, also called drop-off location) for the purpose of analyzing the travel distance and direction. Furthermore, the Euclidean distance, or the direction of residents' travel by taxi, is computed. In this paper, we selected the OD locations as per the following method:

O location: the location where meter state changes from 0 to 1

D location: the location where meter state changes from 1 to 0

Furthermore, we calculated the Euclidean distance by latitude and longitude coordinates for each pair of OD locations. We removed the invalid data (where the value of distance is too large or too small and OD locations are in different time periods). Finally, we have a total of 1,957,470 records of O locations or D locations. The data statistics are shown in Table 1.

Every year, the National Day is celebrated on October 1st to commemorate the founding of the People's Republic of China. Moreover, the seven-day holiday from October 1st through 7th is the so-called "Golden Week." During the Golden Week, more Chinese people travel all over the places. Therefore, in order to analyze the impact of holidays, we divided all data collected in the month of October 2012 into four parts by time (each part including 7 days): (1) Oct 01–Oct 07; (2) Oct 08–Oct 14; (3) Oct 15–Oct 21; (4) Oct 22–Oct 28 as highlighted in Table 1. Here, the first part (Oct 01–Oct 07) can be used to represent the "Golden Week" of National Day.

3. Results and Discussion

3.1. Analysis of Time-Sharing Statistics of OD Locations. For each slot of the seven-day period, we count the number of OD locations for every two hours (such as 0:00–2:00). The recorded statistics are shown in Figure 1. From Figure 1, seven 24-hour cycles can be clearly identified. The identified cycle indicates that the distribution of OD locations is repeated daily. The curves of the number of O locations and the number of D locations of four time periods are very similar: OD locations are mainly at 8:00–12:00 in the morning, at 14:00–20:00 in the afternoon, and at 22:00–0:00 (the next day) in the evening.

Table 1: The data statistics of GPS data records.

	Number of records	Number of taxies	Number of travels	Number of valid travels
Total	387074469	3501	2335582	1957470
National Day	88008559	3501	518800	460783
Other	299065910	3501	1816782	1496687
Weekdays	224907188	3501	1339535	1101645
Weekends	74158722	3501	477247	395042
Oct 01–Oct 07	88008559	3501	518800	460783
Oct 08–Oct 14	87638513	3501	518030	454244
Oct 15–Oct 21	86824240	3501	533341	468382
Oct 22–Oct 28	87023884	3501	541062	469045

Table 2: Akaike weights chosen in the AIC model for displacement distribution.

Data sets	Exponential cut-off power-law distribution	Lognormal distribution	Weibull distribution	Exponential distribution
Oct 01–Oct 07	0.0000	0.4778	0.0000	0.5222
Oct 08–Oct 14	0.0000	0.9786	0.0000	0.0214
Oct 15–Oct 21	0.0000	0.9983	0.0000	0.0017
Oct 22–Oct 28	0.0000	1.0000	0.0000	0.0000

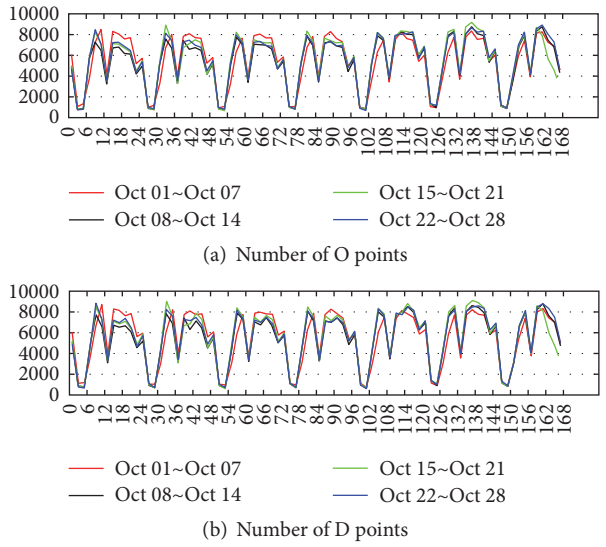


Figure 1: Statistics of OD location numbers at different times. (a) Number of O locations. (b) Number of D locations.

However, the four curves of OD locations also have some nuances: the curves indicate main concentration between 10:00 and 12:00 in the morning during the holidays (the red curve) and weekends (the last two cycles of the black, green, and blue curve). But, the concentration remains between 8:00 and 10:00 in the morning during weekdays (the first five cycles of the black, green, and blue curve). This shows that urban residents' morning travel by taxi during weekdays is earlier than during the holidays and during weekends in general. Moreover, for the time period from 14:00 to 20:00 in the afternoon, the number of OD locations fluctuates around 8000, while during weekdays (the first five cycles of the black, green, and blue curves) the number of OD locations fluctuates around 7000 as indicated by the holiday period (the red curve) and weekends (the last two cycles of the black, green, and blue curves). It also confirms that the urban residents travel by taxis less on weekdays than during the holidays for the afternoon timings.

3.2. Analysis of Spatial Distribution of OD Locations. Mobile step length is an important metric of the mobility. Spatial displacement is commonly used as the mobile step length in the studies on the human mobility. This is because displacement can represent mobility behavior without being affected by the details of paths [21, 22].

Figure 2 shows the distribution of displacement by taxis for residents in Tianjin. However, it is not possible to describe the displacement distribution by the power-law distribution only. So, the current paper provides a comparison of four distributions, namely, power-law distribution with an exponential cutoff (PLEXP), lognormal distribution (LN), Weibull distribution (WB), and exponential distribution (EXP). The distributions are represented by the red dotted line, blue solid line, green dotted line, and carmine dotted line, respectively. As per results depicted in Table 2 and following Akaike information criterion (AIC), it can be concluded that the displacement distribution for taxi passengers follows the lognormal distribution and the exponential distribution. Table 3 presents the optimal fitting parameters. Another interesting phenomenon is that the displacement distribution can be partitioned into two parts at 20 km. The first part had a slow increase and then sustained a stable decrease. There was an obvious peak in the second part.

It can be observed from Figure 2 that increase in displacement leads to first increase of the displacement distribution density function $P(\Delta r)$ to a high level which then decreases slowly for traveling behavior with a displacement of less than

Table 3: Result of optimal fitting on displacement distribution.

Data sets	μ (lognormal distribution)	μ (lognormal distribution)	λ (exponential distribution)
Oct 01–Oct 07	1.321 (1.307, 1.334)	0.8683 (0.8522, 0.8845)	0.2146 (0.2109, 0.2184)
Oct 08–Oct 14	1.178 (1.171, 1.186)	0.6716 (0.6635, 0.6796)	0.3073 (0.2950, 0.3197)
Oct 15–Oct 21	1.174 (1.166, 1.181)	0.6783 (0.6702, 0.6863)	0.3073 (0.2956, 0.3190)
Oct 22–Oct 28	1.083 (1.069, 1.097)	0.7949 (0.7793, 0.8105)	0.2942 (0.2894, 0.2990)
Oct 01–Oct 07	1.321 (1.307, 1.334)	0.8683 (0.8522, 0.8845)	0.2146 (0.2109, 0.2184)

Parameters (confidence interval of 95%).

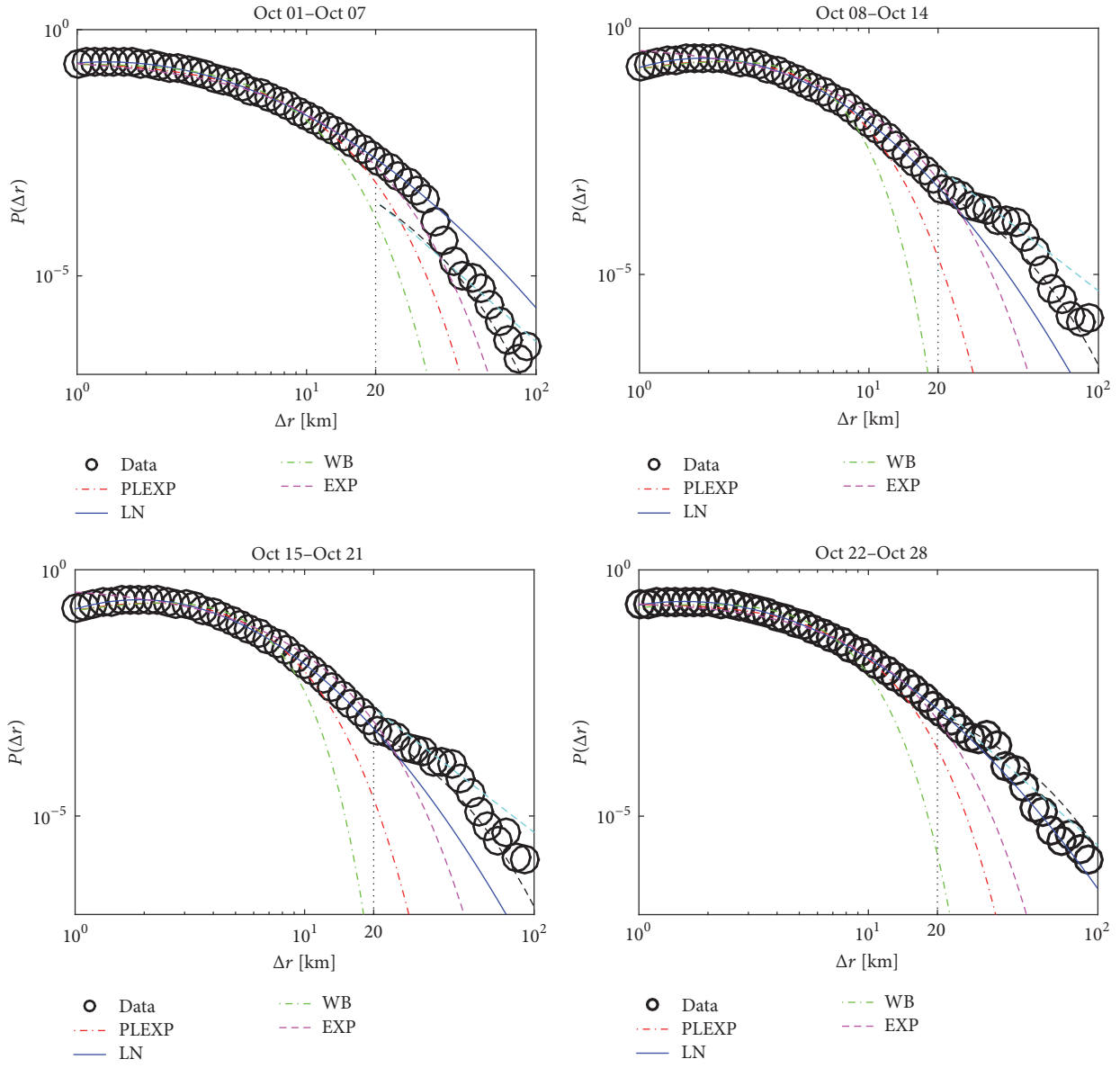


Figure 2: Distribution of displacement by taxi.

20 km. $P(\Delta r)$ reaches its peak value at Δr = rm, where rm varies with time and ranges from 1.3 to 2.3 km. It is easy to understand the increase in $P(\Delta r)$, as residents prefer to travel on foot or by bike for distances less than 1 km. Moreover, rm also indicates that people take traveling costs into account in their daily life.

Analysis of the data sets over the four time periods reveals that 97% of passengers have a displacement of less than 20 km. This coincides with experience collected from the daily life. Considering travel costs, residents prefer to travel long distances on public transport or by private car, rather than by taxi, during holidays.

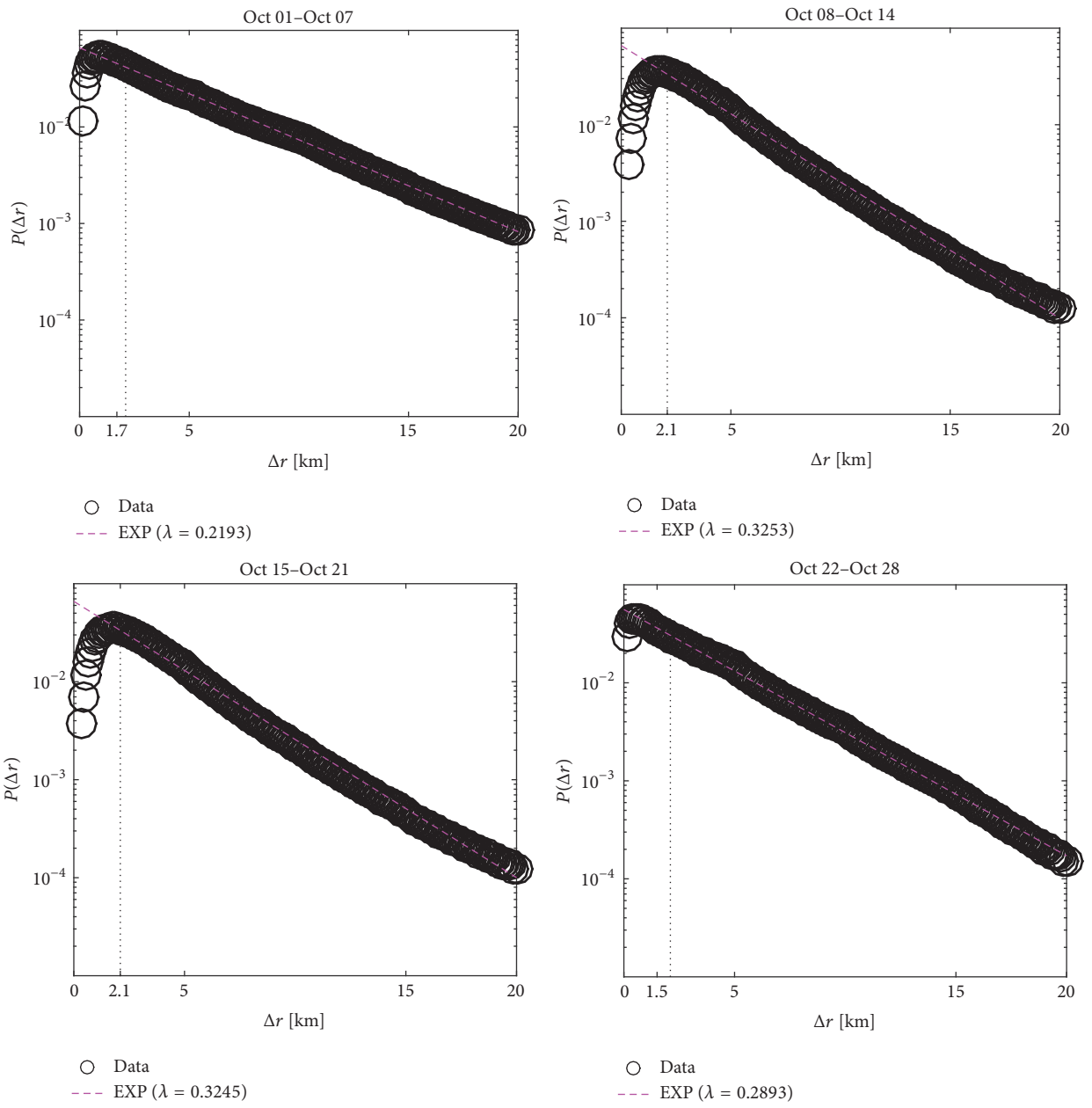


Figure 3: Distribution of displacement for short-distance travels.

From Figure 3, it can be deduced that the traveling behaviors with a displacement in the rm-20 km range decreased exponentially and its corresponding parameters are as given in Table 4. The values indicate that the current part accounts for more than 67% of the travel distance of all passengers. Here, the value parameter λ varies with time, but its values remain to be close to each other. In a nutshell, the displacements of residents traveling by taxi in different cities share similar statistical features. All trips above 2 km follow the two-piece exponential distribution. The distribution during National Day does not differ greatly from the distribution during regular workdays.

Table 4: Results achieved by fitting displacement in piecewise exponential distribution.

Data sets	λ (the first part)	λ (the second part)
Oct 01–Oct 07	0.2193 (0.2181, 0.2205)	0.1628 (0.1473, 0.1783)
Oct 08–Oct 14	0.3253 (0.3204, 0.3303)	0.1066 (0.0799, 0.1332)
Oct 15–Oct 21	0.3245 (0.3198, 0.3292)	0.1103 (0.0841, 0.1366)
Oct 22–Oct 28	0.2893 (0.2872, 0.2914)	0.0741 (0.0560, 0.0921)

3.3. Analysis of Residents Travel Distance. In order to facilitate the statistical spatial distribution of OD locations, we divide

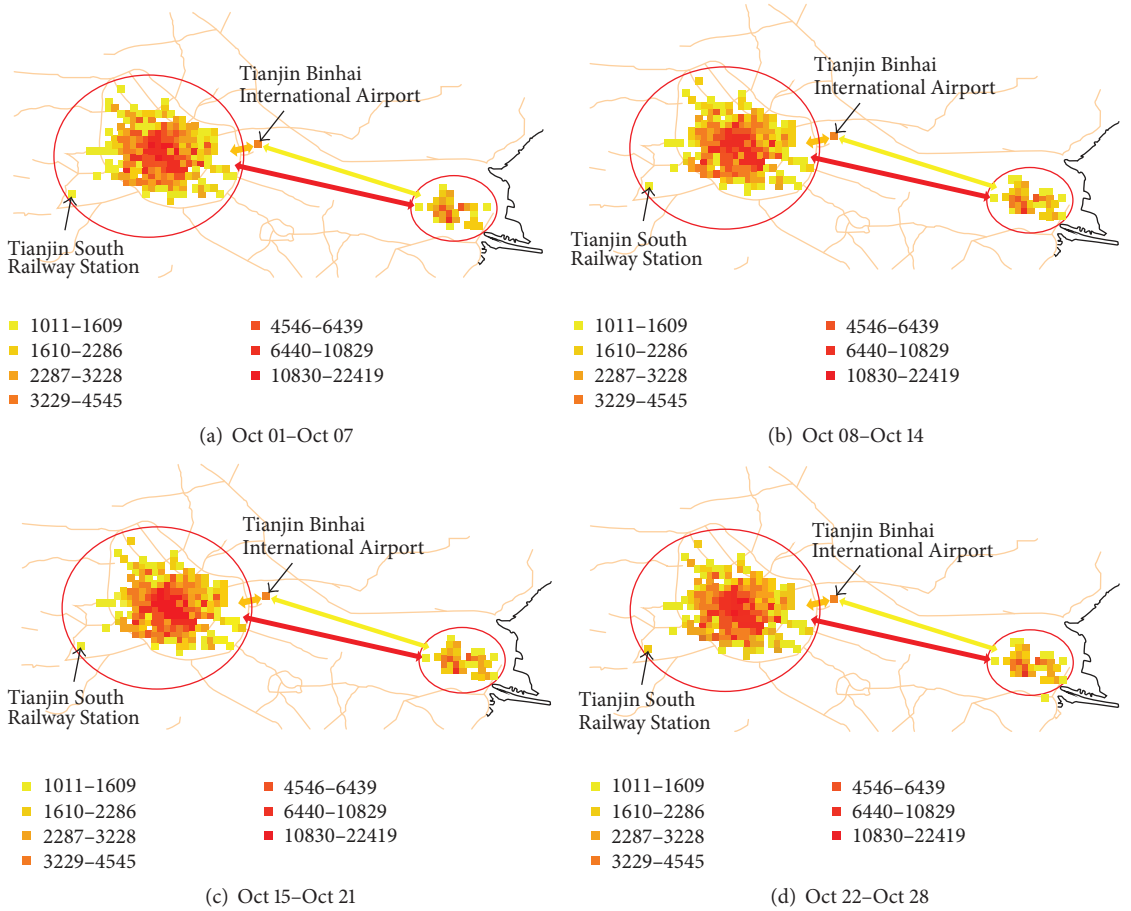


Figure 4: Spatial distribution of OD locations during (a) the first part of October (Oct 01–07), (b) the second part of October (Oct 08–14), (c) the third part of October (Oct 15–21), and (d) the fourth part of October (Oct 22–28).

the map into different grids (the resolution of each grid is 0.01 longitude and 0.01 latitude), count the number of OD locations in each grid, and visually indicate the grids having an OD location number greater than 1000 in the map as depicted in Figure 4.

It can be observed from Figure 4 that the spatial distribution of OD locations in the four different slots of study period is similar mainly in the Tianjin urban area (large gathering area in the red ellipse on the left), Binhai New Area (red ellipse on the right beside the sea), and the two isolated locations of the Tianjin Binhai International Airport (orange grid on the right of Tianjin urban area) as well as Tianjin South Railway Station (yellow grid on the bottom left of Tianjin urban area). Out of the specified areas, the airport is a hot area having a number of OD locations of around 4400, which accounts for about 4.7% of the total OD locations. The distribution of OD locations depicted in Figure 4 also reflects the geographic characteristics of Tianjin: "Two City," namely, the Tianjin urban area and Binhai New Area.

Travel distance is an important measurement in describing travel behavior. It is measured by calculating the Euclidean distance of the respective pairs of OD locations. The probability distributions of the residents' travel distance (D) are shown in Figure 5. From Figure 5, it can be seen that probability curves representing different slots for the month of October 2012 are also very similar. The similarity here indicates that residents' travel distance by taxis is not affected by holidays.

Here, in Figure 5, the red curve represents the residents' travel distance distribution during National Day; the other three curves (black, green, and blue) represent residents' travel distance distribution on rest of the days. We can observe that there is no significant difference in the four curves that indicates the null effect upon residents' taxi travel distance of the holidays. In order to measure the similarity between two probability distributions, Hellinger distance [23–26] is one of the most commonly used metrics. So, we use it to compare the similarity between the distributions of travel distance. The Hellinger distance for measuring similarity between continuous probability functions $p(x)$ and $q(x)$ over a domain X is defined as follows [27]:

$$\mathrm{DH} = \int \sqrt{P(x)\, q(x)}. \tag{1}$$

Table 5: Hellinger distance of travel distance distribution.

	Oct 01–Oct 07	Oct 08–Oct 14	Oct 15–Oct 21	Oct 22–Oct 28
Oct 01–Oct 07	1	0.999817	0.999718	0.999756
Oct 08–Oct 14	0.999817	1	0.999901	0.999926
Oct 15–Oct 21	0.999718	0.999901	1	0.999925
Oct 22–Oct 28	0.999756	0.999926	0.999925	1

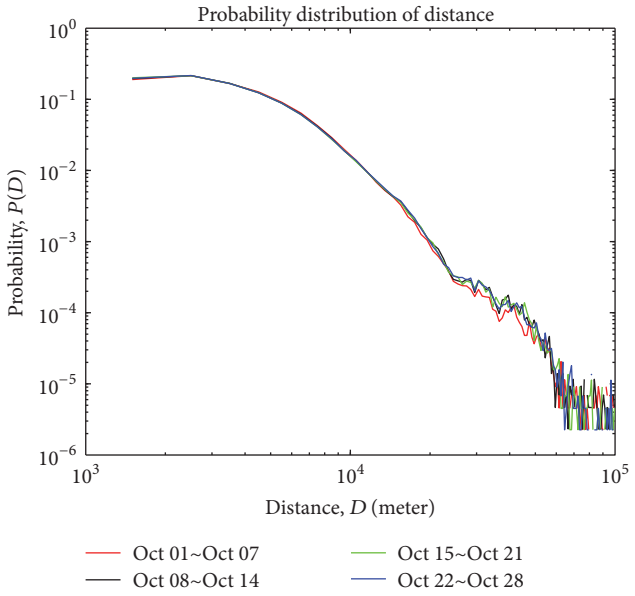


Figure 5: Probability distribution of distance.

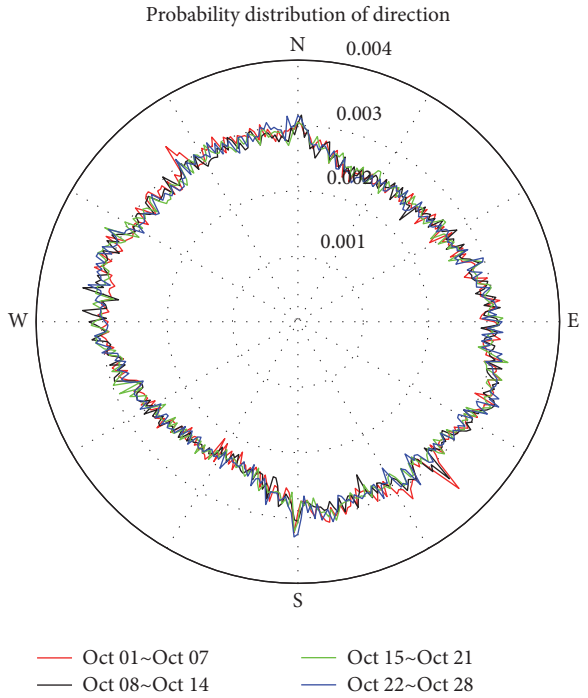


Figure 6: The probability distribution of residents' travel direction.

For discrete distributions, the Hellinger distance is computed as follows:

$$\mathrm{DH} = \sum_{x \in X} \sqrt{p(x)\, q(x)}. \quad (2)$$

We computed the Hellinger distance of travel distributions over the four slots for the study period and listed the values in Table 5. The lower left and upper right of Table 5 are perfectly symmetric, because the Hellinger distances between $p(x)$ and $q(x)$ and vice versa are same. It can be observed from Table 5 that the Hellinger distance between every pair of the four parts is greater than 0.999. It indicates the high similarity between the mobility patterns of any two parts. However, the Hellinger distance between the National Day and the other three periods of time is about 0.9998. Whereas, the Hellinger distance between every pair of the other three parts is higher than 0.9999. This is a nuanced difference that is reflected by the red probability curve in Figure 3. The curve represents its lowest value in comparison to the other three curves after 20 km as described in Figure 3. Therefore, it indicates that long-distance travel by taxi has declined during the holiday period.

3.4. Analysis of Residents Travel Directions. The statistical distribution of travel directions is also an important measure in describing travel behavior. Each trip can be represented as a vector (also called an OD vector) in space.

The distribution of the residents' travel directions during the four slots of the study periods in a polar coordinate system is as depicted in Figure 6. It can be observed from Figure 6 that the overall distribution of the travel directions of the four slots for the study period is very similar and has central symmetry. This shows that the movements of the vast majority individuals can be considered round travel, such as leaving home to go to work in the morning and coming home from work in the evening. The elliptical nature of the curve indicates the uneven distribution of travel direction. It signifies that travel in the northwest and southeast directions is more frequent than travel in the northeast and southwest directions.

We also calculate the Hellinger distance of travel direction between the respective pairs of the four slots in the study period and computed values are as shown in Table 6. It can be observed from Table 6 that its lower left and upper right parts are also symmetrical and each value is greater than 0.999. It indicates a high similarity between the mobility patterns of each pair of both the parts. These similarities are more significant than those of travel distance to deduce that

Table 6: Hellinger distance of travel direction distribution.

	Oct 01–Oct 07	Oct 08–Oct 14	Oct 15–Oct 21	Oct 22–Oct 28
Oct 01–Oct 07	1	0.999741	0.999717	0.999708
Oct 08–Oct 14	0.999741	1	0.999777	0.999756
Oct 15–Oct 21	0.999717	0.999777	1	0.999791
Oct 22–Oct 28	0.999708	0.999756	0.999791	1

the regularity of people traveling by taxi is not affected by holidays.

3.5. Analysis of Residents' Travel during Weekends. "Weekends" usually refers to both Saturday and Sunday. At this time, adults and children usually are at rest, and movement might be more diverse than usual period (i.e., weekdays). In this paper, we divided the study days into two parts as described below.

Weekdays (all the weekdays from Oct 8 to Oct 31)

Weekends (all the weekends from Oct 8 to Oct 31)

We compared the distributions of travel distances and travel directions between these two slots of the study period for weekdays and weekends as depicted in Figure 7. It can be observed that travel distance and travel direction by taxi are very similar on weekends and weekdays. This proves that the regularity of human travel by taxi is not affected by holidays.

3.6. Analysis of the Scope of Activities of Taxis. The scope of taxi's activities can be represented by its radius of gyration and is defined as follows [3, 28]:

$$r_g^a(t) = \sqrt{\frac{1}{n_c^a(t)} \sum_{i=1}^{n_c^a} \left(r_i^{\vec{a}} - r_{\text{cm}}^{\vec{a}}\right)^2}, \tag{3}$$

where $r_i^{\vec{a}}$ represents $i = 1, \ldots, n_c^a(t)$ positions recorded for user a and $r_{\text{cm}}^{\vec{a}} = 1/n_c^a(t) \sum_{t=1}^{n_c^a} r_t^{\vec{a}}$ is the center of mass of the trajectory [29].

In order to compute the scope of taxi's activities, it requires preprocessing of the values. We compute the radius of gyration of each taxi every day in the month of October, and then we screen taxis to make sure that their radius of gyration is larger than 1000 m (a radius of gyration of less than 1000 meters is regarded as invalid data in this study). This makes consider 2198 taxis and calculate their radius of gyration for the four slots of the study period. The computed values for their statistical distributions are plotted as depicted in Figure 8.

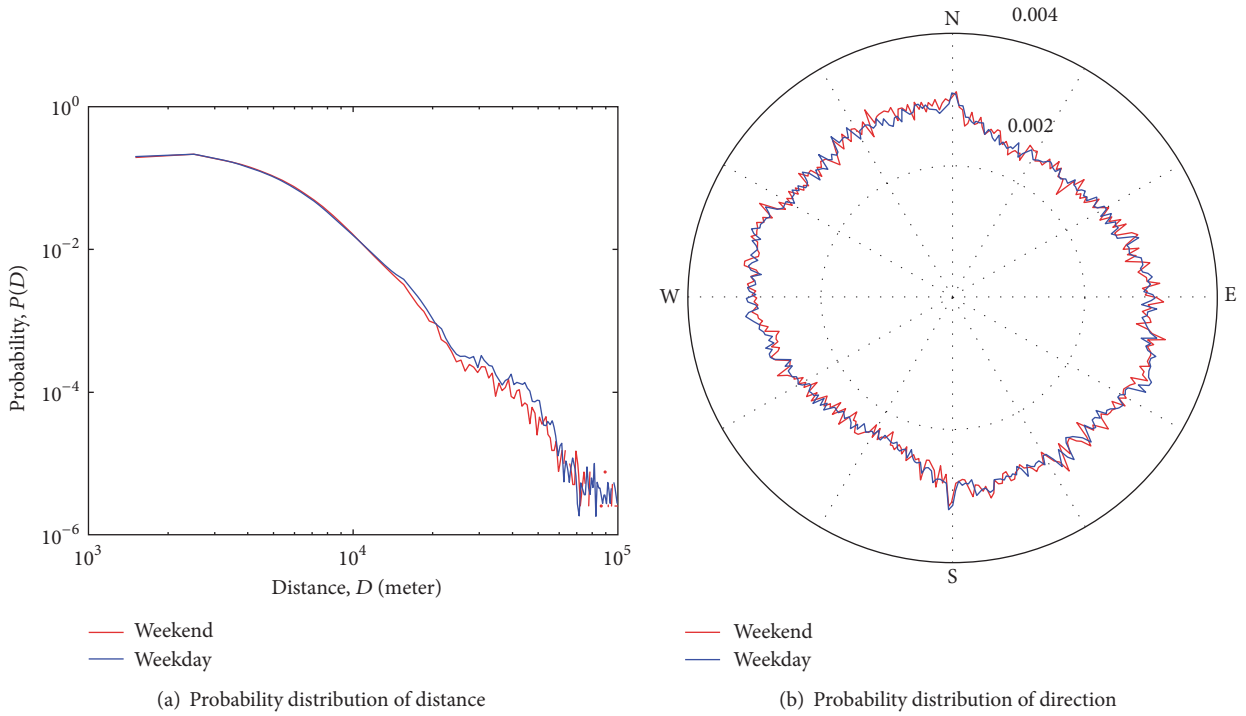
It can be observed that the value of the radius of gyration greater than 10 km on National Day (the red line) is significantly higher than the other three periods. This indicates that, during National Day, most taxis have a wider range of activities. The other three curves (black, green, blue) are very close but not exactly the same, which also shows (1) the diversity of the taxis' movements and (2) that people's activities tend to be steady during ordinary times [30].

3.7. Taxi Mobility throughout the Year. We studied the data set of GPS tracks from the taxi company for the daily tracks of 4,252 taxis in 2012 for obtaining characteristics of taxi mobility throughout the year. We considered over 4.5 billion GPS sample points. These tracks cover the entire city and are concentrated in the central urban zone. GPS sample points were taken at an interval of 24 seconds, and each sample point includes the serial number of the taxi, a time stamp, longitude, latitude, speed, and the number of passengers in the vehicle. We extracted over 25 million cases of passengers traveling by taxi. These passengers had an average stay of 13.6 minutes inside the vehicle and an average displacement of 4.2 km.

The number of passengers traveling by taxi in 2012 is depicted in Figure 9. It can be observed that the residents travel by taxi at a regular weekly interval. Some abnormal points exist where the number of travelers was extremely low. We analyzed the facts behind the abnormalities and that these anomalies happen on special days (e.g., the Chinese New Year's Eve on January 22) or during terrible weather conditions. In particular, urban traffic was low during the rainstorm on July 26 and found to one-third of the average daily level. However, there was little influence on traveling behavior on ordinary holidays such as the National Day.

4. Conclusions

In this paper, we analyzed urban resident travel behavior patterns in Tianjin by using taxi GPS data. Analyzing taxi mobility in the cities during the holidays enables us to study the behavioral patterns of the people. The analysis of the current study will yield good insights for the administration of transportation needs. We analyzed the impact of the spatial distribution of urban residents' pick-up and drop-off locations by taxi, their travel distance and travel direction, and the taxis' scope of activities during the holidays. Based upon results, we concluded the following: (1) the holidays do not affect the spatial distribution of residents' pick-up and drop-off locations by taxi, travel distance, or travel direction; (2) human travel behavior tends to have a stable regularity; (3) during the holidays, taxis have a larger scope of activities, which may be associated with the city's economy as well as taxi operating patterns. The research contribution cited above will help the transportation administration to allocate appropriate resources during the holidays. However, residents often choose taxis as a way to travel in addition to using other means of transportation, such as walking, buses, and cars, whereas, in addition to GPS data, many other sources like mobile phone records can also represent human movement. Future research in the field should be focused



(a) Probability distribution of distance

(b) Probability distribution of direction

FIGURE 7: (a) The probability distribution of residents' travel distance between weekdays and weekends. (b) The probability distribution of residents' travel direction between weekdays and weekends.

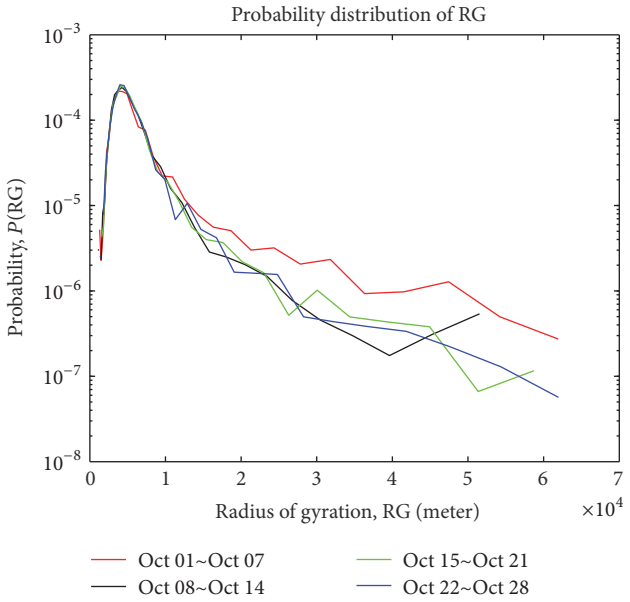


FIGURE 8: The probability distribution of radius of gyration.

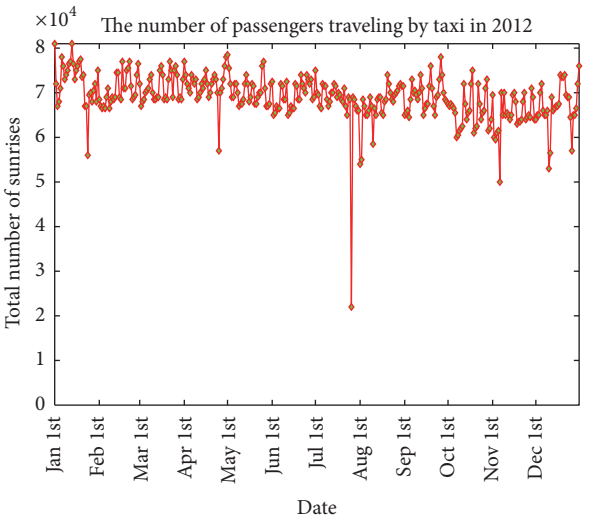


FIGURE 9: The number of passengers traveling by taxi in 2012.

on taking these factors into consideration for analyzing and predicting the human travel behavior.

Conflicts of Interest

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

Acknowledgments

This research is partially supported by the Major Project of National Social Science Fund of China (14ZDB153), the National Science Foundation of China (61572355), the Tianjin Research Program of Application Foundation and Advanced Technology (15JCYBJC15700), and the Fundamental Research of Xinjiang Corps (2016AC015).

References

[1] A.-L. Barabási, "The origin of bursts and heavy tails in human dynamics," *Nature*, vol. 435, no. 7039, pp. 207–211, 2005.

[2] D. Brockmann, L. Hufnagel, and T. Geisel, "The scaling laws of human travel," *Nature*, vol. 439, no. 7075, pp. 462–465, 2006.

[3] M. C. Gonzalez, C. A. Hidalgo, and A. L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, pp. 779–782, 2008.

[4] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[5] M. Barthélemy, "Spatial networks," *Physics Reports*, vol. 499, no. 1–3, pp. 1–101, 2011.

[6] B. Jiang, J. Yin, and S. Zhao, "Characterizing the human mobility pattern in a large street network," *Physical Review E*, vol. 80, no. 2, Article ID 021136, 2009.

[7] C. Song, T. Koren, P. Wang, and A.-L. Barabási, "Modelling the scaling properties of human mobility," *Nature Physics*, vol. 6, no. 10, pp. 818–823, 2010.

[8] C. Roth, S. M. Kang, M. Batty, and M. Barthélemy, "Structure of urban movements: polycentric activity and entangled hierarchical flows," *PLoS ONE*, vol. 6, no. 1, Article ID e15923, 2011.

[9] B. Jiang and T. Jia, "Exploring human mobility patterns based on location information of US flights," 2011, https://arxiv.org/abs/1104.4578.

[10] X. Y. Yan, X. P. Han, B. H. Wang, and T. Zhou, "Diversity of individual mobility patterns and emergence of aggregated scaling laws," *Scientific Reports*, vol. 3, no. 2678, 2013.

[11] G. Pan, G. Qi, Z. Wu, D. Zhang, and S. Li, "Land-use classification using taxi GPS traces," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 113–123, 2013.

[12] A. Wesolowski, N. Eagle, A. J. Tatem et al., "Quantifying the impact of human mobility on malaria," *Science*, vol. 338, no. 6104, pp. 267–270, 2012.

[13] X. Lu, L. Bengtsson, and P. Holme, "Predictability of population displacement after the 2010 Haiti earthquake," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 29, pp. 11576–11581, 2012.

[14] X. Liang, J. Zhao, L. Dong, and K. Xu, "Unraveling the origin of exponential law in intra-urban human mobility," *Scientific Reports*, vol. 3, article 2983, 2013.

[15] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the levy-walk nature of human mobility," *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 630–643, 2011.

[16] X. Liang, X. Zheng, W. Lv, T. Zhu, and K. Xu, "The scaling of human mobility by taxis is exponential," *Physica A: Statistical Mechanics and Its Applications*, vol. 391, no. 5, pp. 2135–2144, 2012.

[17] C. Peng, X. Jin, K.-C. Wong, M. Shi, and P. Liò, "Collective human mobility pattern from taxi trips in urban area," *PLoS ONE*, vol. 7, no. 4, Article ID e34487, 2012.

[18] S. Rambaldi, A. Bazzani, B. Giorgini, and L. Giovannini, "Mobility in modern cities: looking for physical laws," *Proceedings of the ECCS*, vol. 7, p. 132, 2007.

[19] A. Bazzani, B. Giorgini, S. Rambaldi, R. Gallotti, and L. Giovannini, "Statistical laws in urban mobility from microscopic GPS data in the area of Florence," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2010, no. 5, Article ID P05001, 2010.

[20] G. Riccardo, B. Armando, and R. Sandro, "Towards a statistical physics of human mobility," *International Journal of Modern Physics C*, vol. 23, Article ID 1250061, 2012.

[21] J. Yin, W. Lo, S. Deng, Y. Li, Z. Wu, and N. Xiong, "Colbar: a collaborative location-based regularization framework for QoS prediction," *Information Sciences*, vol. 265, pp. 68–84, 2014.

[22] Y. Yin, S. Aihua, G. Min, X. Yueshen, and W. Shuoping, "QoS prediction for web service recommendation with network location-aware neighbor selection," *International Journal of Software Engineering and Knowledge Engineering*, vol. 26, no. 4, pp. 611–632, 2016.

[23] J. Vegelius, S. Janson, and F. Johansson, "Measures of similarity between distributions," *Quality and Quantity*, vol. 20, no. 4, pp. 437–441, 1986.

[24] E. Torgersen, *Comparison of Statistical Experiments*, vol. 36, Cambridge University Press, 1991.

[25] Z. Xia, X. Wang, X. Sun, and B. Wang, "Steganalysis of least significant bit matching using multi-order differences," *Security and Communication Networks*, vol. 7, no. 8, pp. 1283–1291, 2014.

[26] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340–352, 2016.

[27] S. Pang, T. Ma, and T. Liu, "An improved ant colony optimization with optimal search library for solving the traveling salesman problem," *Journal of Computational and Theoretical Nanoscience*, vol. 12, no. 7, pp. 1440–1444, 2015.

[28] S. Pang, C. Lin, M. Zhou, and Y. Li, "A workflow decomposition algorithm based on invariants," *Chinese Journal of Electronics*, vol. 20, no. 1, pp. 1–5, 2011.

[29] S. Pang, Y. Li, H. He, and C. Lin, "A model for dynamic business processes and process changes," *Chinese Journal of Electronics*, vol. 20, no. 4, pp. 632–636, 2011.

[30] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Information Sciences*, vol. 295, pp. 395–406, 2015.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 2867127, 13 pages
https://doi.org/10.1155/2017/2867127

Research Article

Design of Optimized Multimedia Data Streaming Management Using OMDSM over Mobile Networks

Byungjoo Park,[1] Ankyu Hwang,[2] and Haniph Latchman[3]

[1]*Department of Multimedia Engineering, Hannam University, 133 Ojeong-Dong, Daejeok-Gu, Daejeon 306-791, Republic of Korea*
[2]*Technical R&D Center JVG, 53, Neungwolro No. 10-Gil, Yongin-Si, Gyeonggi-Do, Republic of Korea*
[3]*Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32603, USA*

Correspondence should be addressed to Byungjoo Park; bjpark@hnu.kr

Received 28 October 2016; Revised 15 February 2017; Accepted 28 February 2017; Published 20 March 2017

Academic Editor: Jaegeol Yim

Mobility management is an essential challenge for supporting reliable multimedia data streaming over wireless and mobile networks in the Internet of Things (IoT) for location-based mobile marketing applications. The communications among mobile nodes for IoT need to have a seamless handover for delivering high quality multimedia services. The Internet Engineering Task Force (IETF) mobility management schemes are the proposals for handling the routing of IPv6 packets to mobile nodes that have moved away from their home network. However, the standard mobility management scheme cannot prevent packet losses due to longer handover latency. In this article, a new enhanced data streaming route optimization scheme is introduced that uses an optimized Transmission Control Protocol (TCP) realignment algorithm in order to prevent the packet disordering problem whenever the nodes in the IoT environment are communicating with each other. With the proposed scheme, data packets sequence realignment can be prevented, the packet traffic speed can be controlled, and the TCP performance can be improved. The experimental results show that managing the packet order in proposed new scheme remarkably increases the overall TCP performance over mobile networks within the IoT environment thus ensuring the high quality of service (QoS) for multimedia data streaming in location-based mobile marketing applications.

1. Introduction

The Internet of Things (IoT) has been rapidly evolving which has changed our way of living allowing us to innovate new designs and services. IoT provides network architecture for physical objects such as devices, equipment, vehicles, homes, or buildings that are embedded with sensors and actuators. It allows the different objects to interact and communicate with each other and enables them to collect and exchange data. The emerging location-based mobile marketing applications for IoT demand mobility management to ensure the quality of service for multimedia data streaming management over wireless/mobile networks, whereas an important challenge for supporting location-based mobile marketing applications in the Internet of Things is the data packet streaming management over wireless and mobile networks.

Location-based mobile marketing applications are consisting of wireless communication networks, mobile devices (such as personal digital assistants (PDAs), smartphones, and navigation devices), geo-information systems, and location or positioning identification. These applications require the support for seamless mobility management among mobile devices to ensure the high degree of accuracy for location requirements.

The mobility management plays a vital role in achieving a high quality of service (QoS) in multimedia data streaming management in an IoT environment for location-based mobile marketing applications. Therefore, IoT convergence networks and mobility management will be essentially important in transmitting multimedia data packets. With the evolution of IoT environments, mobile devices will be moving frequently to foreign networks. A huge amount of multimedia traffic will be developed due to these frequent movements of mobile devices. Thus, the possibility of packet losses and packet ordering problems would likely happen. In order to provide seamless mobile network which meets the

routing requirements of the location-based mobile marketing applications in IoT, the research community has proposed mobility management schemes [1–3].

In this regard, the need to support mobile nodes in IPv6-based networks has been rapidly evolving. Mobile IPv6 is a standard that provides the mobile nodes (MNs) with mobility management across IP-based wireless networks [4] in an IoT environment.

However, in a TCP error control, the occurrence of temporal time delays caused by handovers cannot be determined for its focus is merely on packet losses due to congestions. Unnecessary measures to prevent congestion are provided by the TCP since these packet losses are considered as congestion indication within handovers on wireless networks [5–7].

A mobile node in the standard Mobile Internet Protocol version 6 (MIPv6) maintains two addresses, that is, a home address (HoA) which is a permanent identification address, and a Care-of Address (CoA) which is a temporary address used for redirecting information in order to perform the packet transmission continuously without disconnection of the network layer. The MN must disconnect with the access router where it is currently connected and attach to the new access router (NAR) whenever it moves to another subnet. A new temporal address defined as the Care-of Address (COA) must be obtained by the MN. This new CoA (NCoA) as well as the HoA needs to be registered by the MN to its home agent (HA) and the correspondent nodes (CNs) it is communicating with. The delay incurred during the movement detection known as the handover latency, the configuration time of the NCoA, and the time consumed for a binding update in order to start the Internet services from the new subnet are essential characteristics that must be analyzed in MIPv6. That is, since the packets that are transmitted from the HA or the CN may be lost during the handover, the improvement of the handover performance of MIPv6 has been aimed by the latest works in order to provide real-time support and prevent delays on traffic flows.

Through the newly defined messages in Fast Mobile Internet Protocol version 6 (FMIPv6) [8], *Router Solicitation for Proxy* and *Proxy Router Advertisement*, the MN can obtain the NCoA before its actual movement to a new subnet. This NCoA is also registered by the MN to its previous AR (PAR) in order to indicate that packets can be forwarded to its NCoA. Thus, it can immediately receive the forwarded packets from its PAR as soon as it moves to the new subnet and connect with a new link. In order to prevent packet losses, buffers may exist in PAR and NAR. Thus, packet losses as well as the handover latency will be reduced with this proposal. However, the disordering packet problem between the packets that are tunneled from the home agent (HA) and the previous access router (PAR) and on the packets that are directly delivered by the CN can be caused by the various features in FMIPv6. The congestion control by the TCP causes the duplicate ACKs (DACKs) as a result of the disordering packets degrading the TCP performance on the transport layer. In addition, useless packet retransmissions from the CN can be induced by disordering packets. An efficient disordering packets solution is difficult to provide in wireless/mobile service applications. Some proposals have been analyzed in order to provide solutions to these problems [9–12]

This paper proposes an optimized multimedia data streaming management algorithm to prevent the packet ordering problem during the handover of mobiles nodes for location-based mobile marketing applications within the IoT environment. This is achieved by applying a new route optimization scheme to the modified access router which can support L2 snoop functions and through the additional of an adapted TCP header format at the HA and CN as the source devices. The remainder of this paper is organized as follows. Section 2 explains the previous works and problems in conventional protocols. Section 3 introduces the proposed realignment algorithm called "OMDSM" in order to increase the TCP performance. This section also discusses the comparison of the data packets sequence in the modified access router (MAR) and the final packet arrival indication from the previous access router (PAR) that requires these modifications. The performance evaluations are shown in Section 4. Finally, the conclusions are presented in Section 5.

2. Related Works

2.1. IETF Standard MIPv6. The basic idea of the standard Mobile IPv6 is to provide a mobile node (MN) with a stationary proxy in the form of a home agent (HA) [4]. The standard handover procedure for Mobile IPv6 (MIPv6) is depicted in Figure 1. The home agent intercepts the packets destined to a mobile node whenever it is away from home and forwards these packets directly to the New Care-of Address (NCoA) of the mobile node through tunneling. The home address is being used as the stationary identifier for the mobile node by the transport layer [13]. Tunneling through the home agent is required as a basic solution resulting to a longer path that leads to a degraded performance. Thereby, a route optimization [1] is included in order to improve its performance. Within the route optimization, in order to modify the handling of the outgoing packets between the mobile node's fixed home address and its NCoA, a binding needs to be discovered by the CN. The mobile node then sends its NCoA to the CN when the route optimization is used through the binding update (BU) messages. The packets that are sent by the CN are then routed to the MN's NCoA once the BU message has been received. However, the CN continues to route the packets to the mobile nodes NCoA through the HA until the BU has been received. Thus, the NAR will disorderly receive these two types of packets.

2.2. IETF Fast Handover for MIPv6 (FMIPv6). The MIPv6 movement detection algorithm and CoA configuration procedure have been replaced by a protocol provided by the proposed FMIPv6 in order to reduce its handover latency. The basic operation of the FMIPv6 [8] is shown in Figure 2. The MN is required by the FMIPv6 to acquire a new CoA at the NAR while still connected in the PAR whenever it attempts to move from its PAR going to the NAR. In addition, a BU message needs to be sent by the MN to its PAR in order that

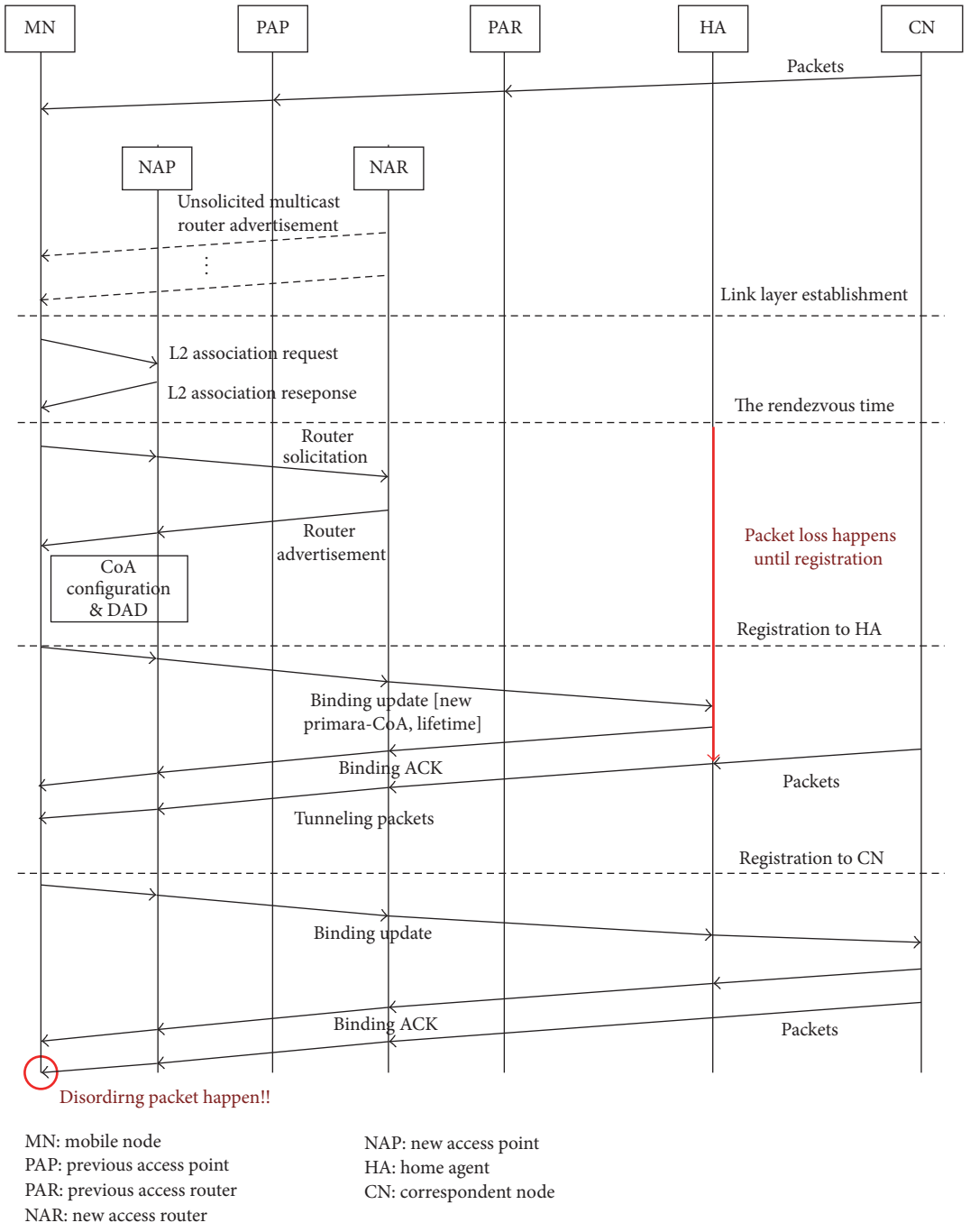


Figure 1: The IETF MIPv6 handover procedure.

its binding cache will be updated with the MN's new CoA. Then, the packets that are originally destined for the MN will be forwarded by the PAR to the NAR. The Fast Handover procedure can be initiated by either the MN or the PAR by using the L2 trigger. The link-layer information indicates the movement of the mobile node (MN) between access routers. An L3 handover will be initiated by the MN by sending a "*Router Solicitation for Proxy* message" to the previous access router (PAR) whenever the MN is receiving an L2 trigger (i.e., mobile-initiated handover). However, when the PAR is the one that received the L2 trigger (Network-controlled handover), a *"Proxy Router Advertisement"* (PrRtAdv) message will be transmitted by the PAR to the suitable MN. An NCoA is obtained by the MN through the network information contained from router advertisements that are broadcasted from the NAR while the MN is still connected to the PAR. The MN's new CoA is validated by the PAR and through the delivery of an HI message to the new access router (NAR); a bidirectional tunnel is formed between the previous access router (PAR) and new access router (NAR). Moreover, a

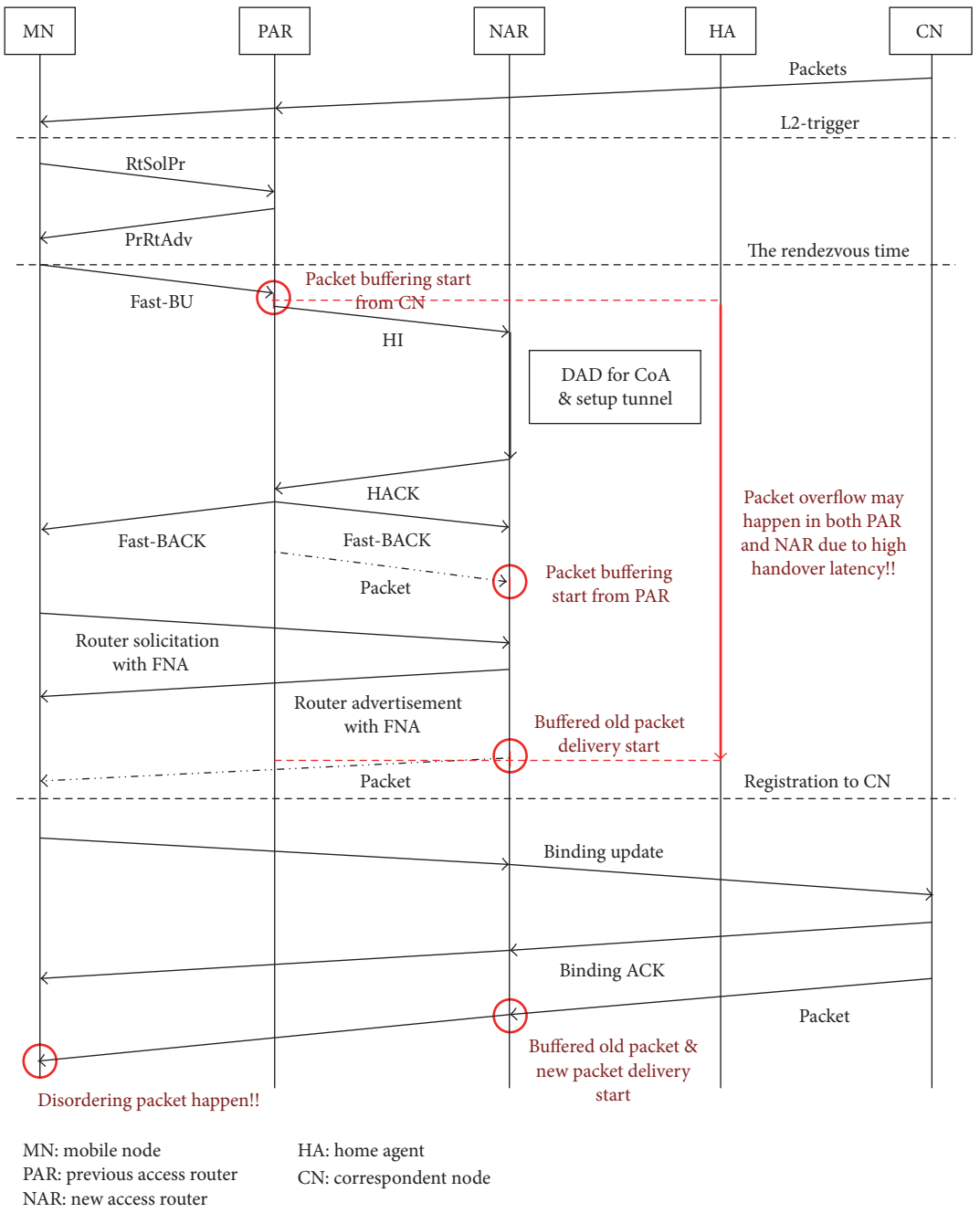


Figure 2: The IETF FMIPv6 handover procedure.

host route is being set up by the new access router (NAR) for the previous Care-of Address (PCoA) of the MN in response to the HA message and a Handover Acknowledge (HACK) message is sent as a reply. A Fast Binding Update (F-BU) should be sent by the MN preferably prior to its disconnection on its link whenever a PrRtAdv message is received. If the FBU message is received by the PAR, wherein the status code in the HACK message indicates, it is required to check if the new access router (NAR) has accepted the handover request. Then, the packets destined to the PCoA on the NAR will be forwarded by the PAR and a Fast Binding Acknowledgement (F-BACK) will be sent to the MN. The MN will then include a Fast Neighbor Advertisement (FNA) option to the Router Solicitation (RS) message that is sent to the new access router (NAR). On the other hand, the NAR includes a Neighbor Advertisement Acknowledgement (NAACK) option to the Router Advertisement (RA) message to be sent to the mobile node (MN). These two messages are exchanged after the link connectivity with the NAR has been changed. The NAR starts to deliver the buffered packets as soon as the NAR sends an RA message with the NAACK option. These buffered packets are delivered through the

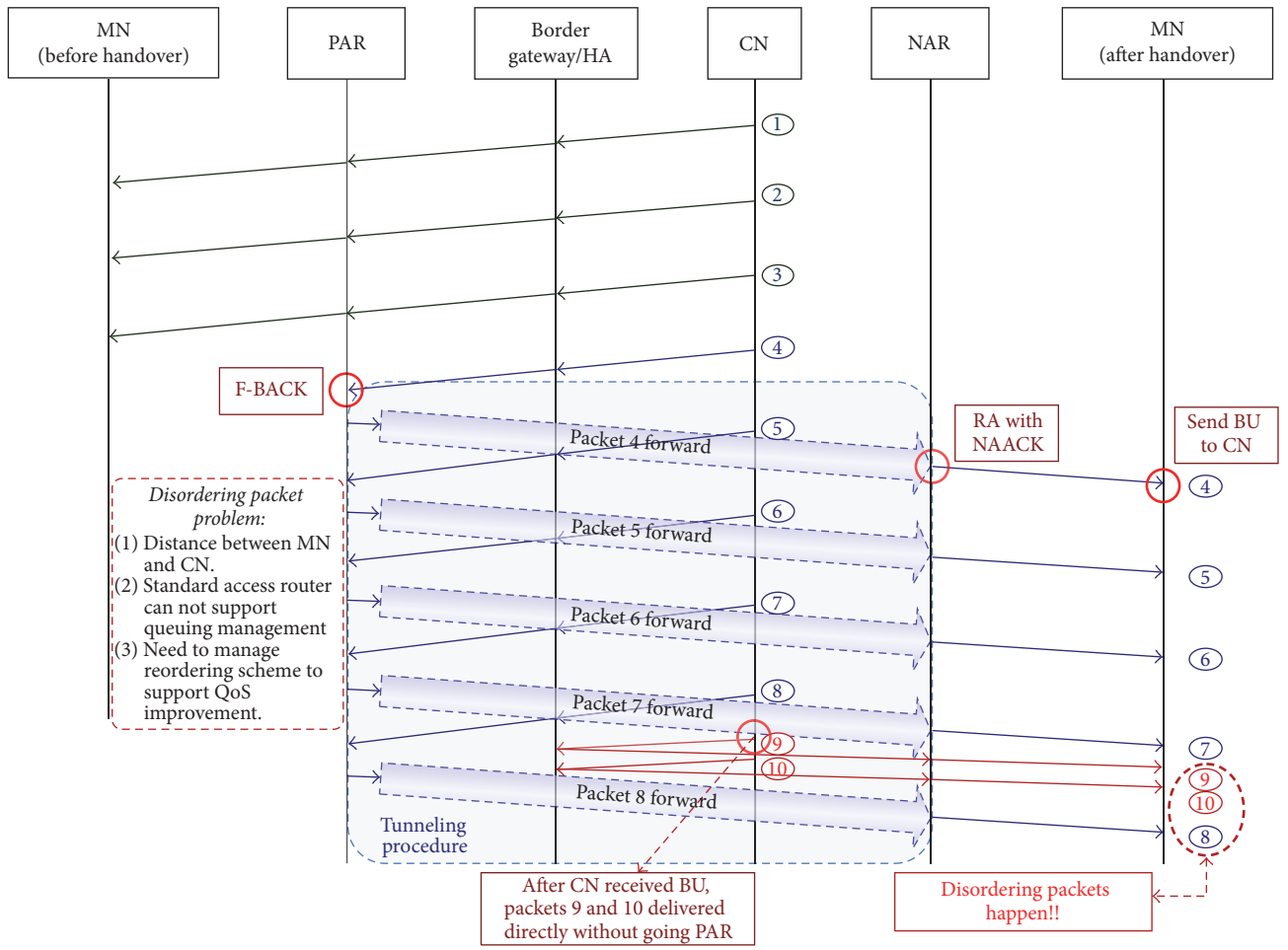
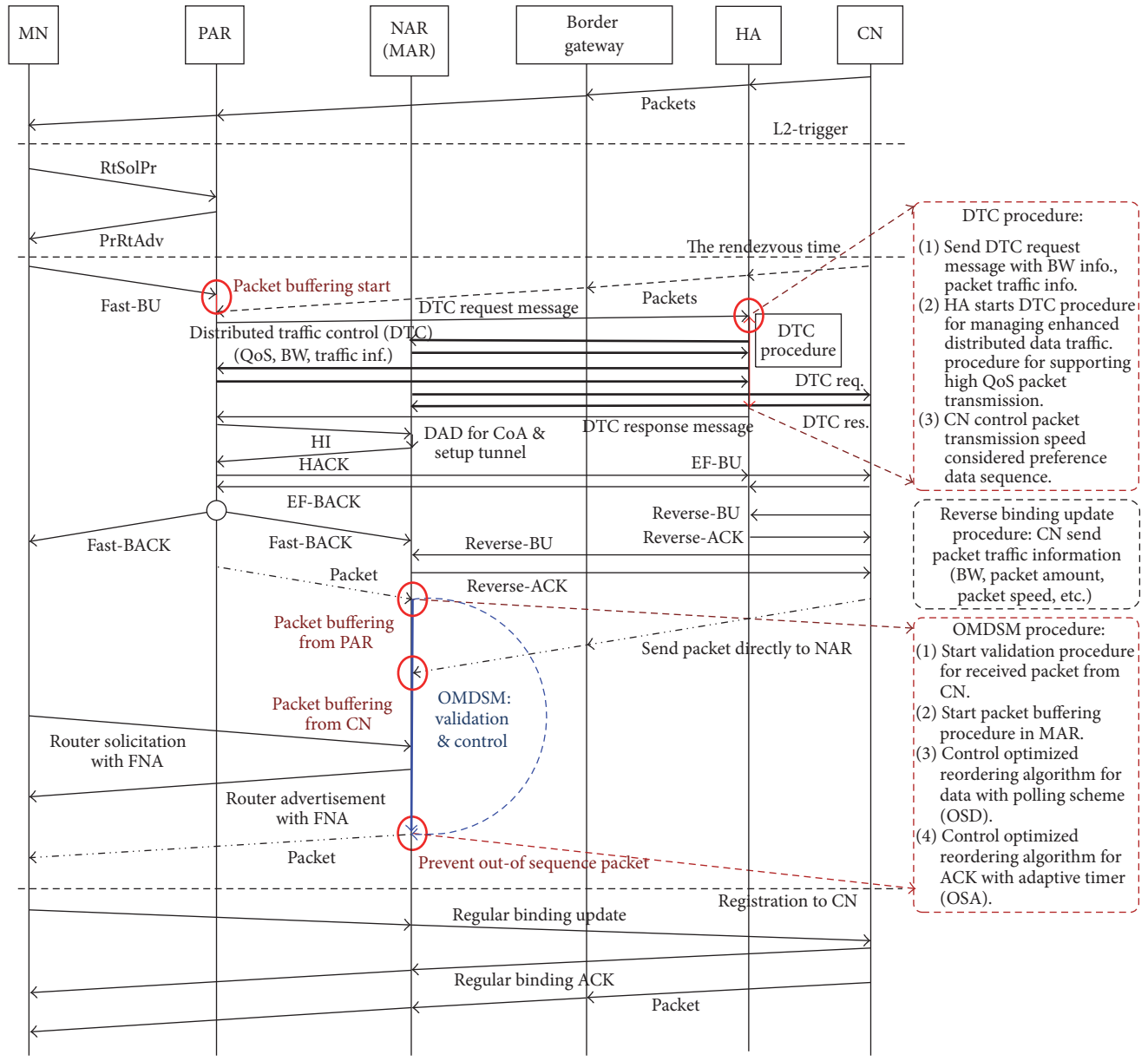


Figure 3: Disordering packet problem in FMIPv6.

bidirectional tunnel from the previous access router (PAR). The packets coming from the correspondent node (CN) are transmitted from the previous access router (PAR) to the new access router (NAR) through a bidirectional tunnel as soon as a binding update (BU) message is received by the CN [14, 15].

The CN can then forward the packets directly to the MN as soon as a BU message is received by the CN. Consequently, the disordered packets may be received by the MN, in the condition that the tunneled distance from the correspondent node (CN) to the new access router (NAR) through the previous access router (PAR) is farther compared to the distance from the correspondent node (CN) to the new access router (NAR). A disordered packet problem example is shown in Figure 3 [16, 17]. Based on the figure, packets four (4) through eight (8) are tunneled from the previous access router (PAR) to the new access router (NAR), wherein it is buffered until a router solicitation (RS) message with a fast neighbor advertisement (FNA) is delivered by the MN to the NAR as soon as an F-BU message is received by the PAR.

The packets nine (9) to ten (10) are sent by the CN to the NAR directly when a BU message from the MN is received by the CN. The new access router (NAR) buffers these packets until the mobile node (MN) receives a router advertisement (RA) with a NAACK option. The new access router (NAR) buffered packets will become disordered because of the packet delay time incurred by the tunneling, that is, whenever it utilizes a tunneling mechanism from the correspondent node (CN) going to the new access router (NAR) through the previous access router (PAR) which is measured to be farther as compared to the transmission from the correspondent node (CN) going to the new access router (NAR) without tunneling. Hence, duplicate ACK (DACK) occurs in the mobile node (MN) for packets seven (7) and eight (8) when an MN receives the disordered packets [18, 19].



MN: mobile node
PAR: previous access router
NAR: new access router/modified AR
HA: home agent
CN: correspondent node

Figure 4: OMDSM handover procedure.

3. OMDSM: Optimized Multimedia Data Streaming Management Algorithm with Traffic Distribution

The performance of TCP in both wired and wireless networks suffers from drawbacks of packet losses caused by bit-errors. This problem was assumed by the TCP sender to be caused by the congestion of the network traffic. Hence, the transmission window of the sender of the TCP is dropped and frequent timeouts occur resulting to a degraded throughput. In order to improve the performance of the TCP, the snoop protocol has been proposed while recovering the wireless errors locally in a wireless LAN environment [12, 20].

In this section, a new data traffic controller scheme is proposed to manage a packet flow which can support a reliable traffic QoS and multimedia packet realignment scheme to enhance the performance of TCP in IP-based wireless networks through the disordered packets elimination throughout the handover process. The proposed OMDSM handover procedure is shown in Figure 4. The model is

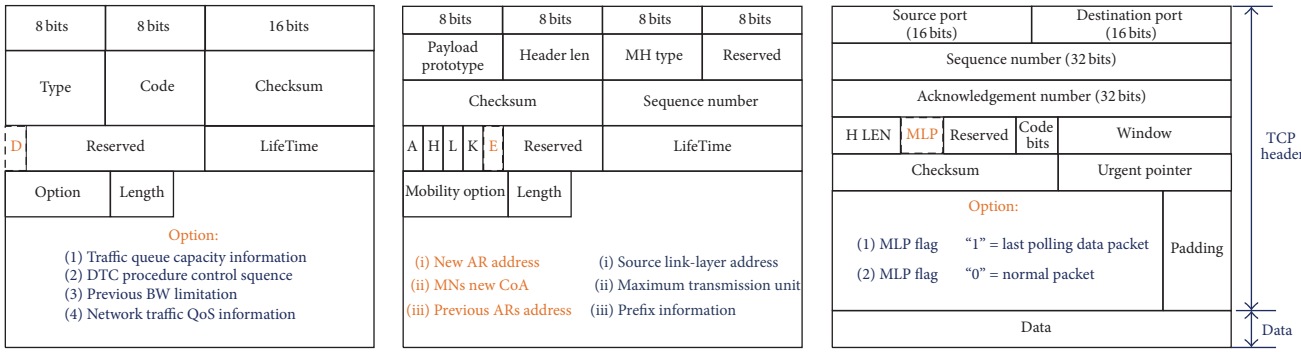


Figure 5: The DTC, EFBU, and MLP message formats: (a) DTC, (b) EFBU, and (c) MLP.

similar to the snoop protocol wherein a duplicate ACK (DACK) is prevented and the sequence of TCP data packets is controlled in the access point (AP).

However, link level snoop functions are only applied into the modified integrated access router. Also, the TCP packet transmission time structure is considered between the data packets and ACK packets which can prevent the disordering of packets using adaptive timer in the modified integrated access router called "MAR." The MAR with snoop agents consists of a controller, a buffer, traffic manager, and a sequence checker. In this article, a snoop agent is implemented with link level buffers in the modified access router (MAR) as specified by the snoop protocol causing the packets that are flowing through the wireless link to be cached. Thus, the unacknowledged packets retransmission can be avoided; hence, the unnecessary timeouts can be prevented. The duplicated packets can also be prevented through the filtering of acknowledgements that are copied. There are two main routines that allow these functions to be performed: the optimized snoop for data (OSD) and snoop for ACK (OSA).

3.1. Data Traffic Control (DTC) Reverse BU Control and Route Optimization. During the movement detection, due to channel maintenance or L3 handover, the handover is performed by the mobile node (MN) to another access point (AP). The list of AP's L2 information will be the result of scanning performed by the MN. The MN sends the association request message with NAP's MAC address as soon as the comparison of previous AP (PAP) L2 power with new AP (NAP) L2 power is done. A scan will then be performed by the MN in order to see the APs through probes. The PAR immediately send a data traffic control (DTC) request message to HA and CN as soon as it receives the Fast-BU message from the MN. The HA broadcasts a DTC message which can support seamless traffic control to all neighboring access routers. In order to support a reliable data traffic distribution without packet interruption, an optional DTC traffic control procedure has been optimized. This can take place through the utilization of a 1-bit D-flag in the reserved field and notifying the node that follows the proposed scheme.

Table 1: The E-Flag of EF-BU message.

E-Flag	Mean
00	EF-BU message does not apply in the case of IEEE 802.
01	The MN's new CoA will be used.
10	Data packets must be sent to the MN's old CoA.
11	The standard BU message needs to be used.

This bit is named as "DTC Request bit (D bit)" wherein it contains four options, namely, "Traffic Capacity Option," "DTC Procedure Option," "Previous BW Limitation," and the "Network Traffic QoS address." Upon receiving a DTC message, ARs and CN can share traffic information with abutting ARs and save a certain period of time to its own buffer. Figure 5(a) shows the formats of the DTC message. The PAR sends a handover initiation (HI) message to the NAR in order to set up the tunnel as soon as the DTC message has been sent. In the proposed scheme, as soon as the establishment process of the bidirectional tunnel from the PAR to the NAR is done, a new enhanced fast binding update (EF-BU) [16] message will be sent by the PAR to the CN. This is done as soon as the MN starts moving in order to decrease the number of packets that are needed to be forwarded from PAR to the NAR. That is, the PAR sends quickly an EF-BU to the CN as soon as the tunnel between the PAR and NAR is setting up. This EF-BU message can be modified by adding a 2-bit E-flag to the reserved flag that includes the "New AR address" and "MNs New CoA" as options in the option field. The formats of the EF-BU message are shown in Figure 5(b) and the E-bits definitions are depicted in Table 1 [16]. The CN has to be operated by the E-bits whenever it receives and EF-BU message. As soon as the EF-BACK message was sent, the CN sends a reverse binding update (reverse-BU) message that include the CN's packet speed, BW, and packet processing priority information.

Each of the message exchanges in the OMDSM scheme is defined as follows:

(1) After receiving a Fast-BU, the PAR sends a DTC request message to HA. The HA starts the DTC procedure and the PAR buffers the packet addressed to PCoA.

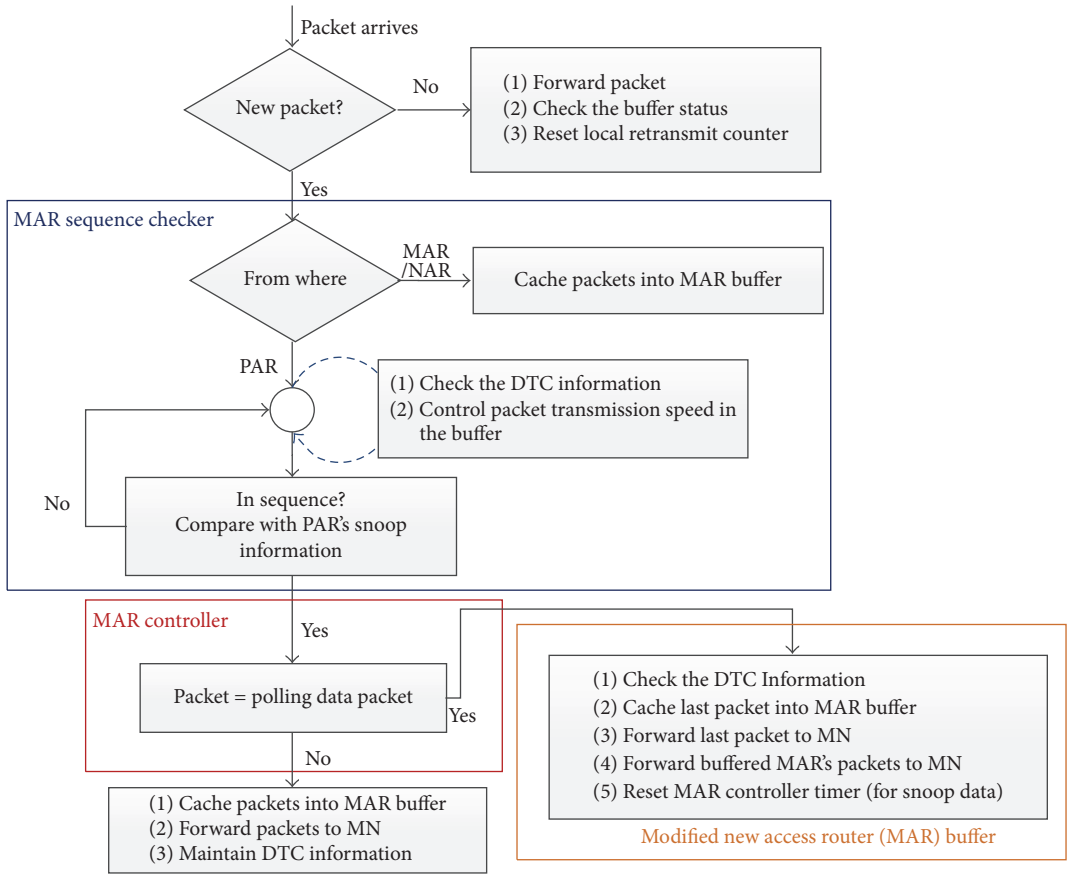


Figure 6: The flow chart of optimized MAR snoop function for data (OSD).

(2) The HA sends the DTC information to all neighboring access routers (ARs).

(3) The PAR sends an EF-BU message after finishing the tunneling-path between PAR and NAR.

(4) The CN sends an EF-BACK message to the PAR. The CN then sends a modified TCP data packet after setting the MLP flag to "1."

(5) At the same time, the CN sends a reverse binding update message to HA and MAR. After the HA received the reverse binding update, it replies with a reverse binding acknowledgement to the CN.

(6) An F-BACK message is then sent by the PAR to the MN and NAR.

(7) The PAR starts to forward the buffered packets to the NAR with an adequate packet transmission speed using the DTC information.

(8) The NAR starts to check the received TCP data packet MLP flag. The MAR starts the OMDSM packet managing procedure which can supply the ordering sequence.

(9) The MN sends a router solicitation message to the NAR.

(10) The NAR sends back a router advertisement message to the MN.

(11) The CN send packets to the MN addressed to the NCoA.

(12) The NAR buffers the packets addressed to NCoA until getting the tunneled packet with an MLP flag "1."

(13) After receiving the last tunneled packet with MLP flag "1," the NAR deliver the buffered packets which came directly from the CN.

3.2. Optimized Realignment Algorithm for Data with Polling Scheme. The realignment algorithm flowchart for data to perform the proposed scheme is shown in Figure 6. Initially, during a handover, a handover initiation (HI) message that includes a snoop information of the previous access router (PAR) is sent in order for the sequence of the TCP packets to be controlled after the PAR has received an F-BU message. The remaining packets are directly sent by the correspondent node (CN) to the new access router (NAR) whenever it receives an EF-BU message. In addition, the CN also sends the last packet with a modified header to the PAR. The buffered packets are directed by the previous access router (PAR) to the new access router. It happens as soon as an F-BACK message is delivered by the previous access router (PAR) going to both the new access router (NAR) and the mobile node (MN). The format of the TCP packet with MLP flag is depicted in Figure 5(c). An MLP flag bit can be added

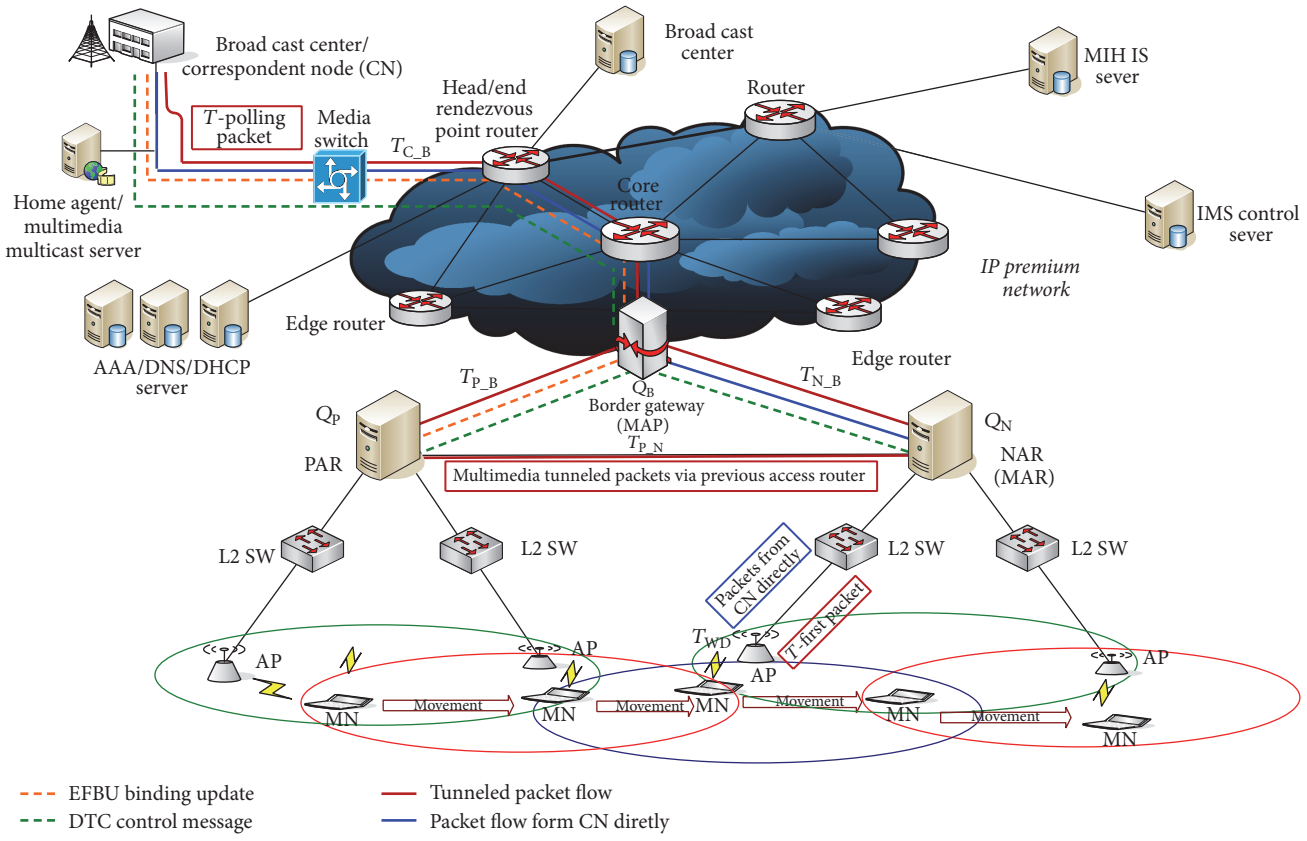


Figure 7: The packet transmission during handover in OMDSM.

to modify the TCP packet reserve field in the TCP header in order for the last packet to be distinguished from all of the TCP packets coming from the new access router (NAR). The last packet that is modified is called the MLP.

Thus, the option field in the TCP packet will be utilized. If the MLP flag is set to "0," then the packet acts as a normal packet. On the other hand, if the MLP flag is set to "1," then the packet is acting as the polling data packet that originates from the previous access router (PAR). Whenever the CN receives an EF-BU message from the PAR, the last data packet as well as a polling data packet is simultaneously sent to the NAR. The polling data packet acts as a control message to the MN signaling where no more tunneled packets exist. The PAR can then remove the MN's information as soon as it receives this polling message. A new data packet can be sent without the tunneling process by the correspondent node (CN) going to the new access router (NAR) after it has sent a polling data packet. At first, the PAR's snoop information that is included in the HI message will be used by the MAR sequence checker for determining whether the packet that is received comes from either of the previous access routers (PAR) of the new access router (NAR). That is, the snoop information about the PAR is included in the HI message whenever it is sent by the PAR to the NAR. Then, the MAR controller starts checking for the MLP flag, that is, to check whether the arriving packets are received in a correct sequence. The MLP flag is essentially important in distinguishing between packets delivered through tunneling from the PAR and those packets that are directly delivered from the CN without tunneling. The MAR buffers the packets that are directly transmitted by the CN until the NAR receives the MLP with the flag bit that is set to "1." The packet transmission network architecture during a handover in OMDSM scheme is shown in Figure 7. As depicted in Figure 7, the packets that are sent directly from the CN to NAR are cached in the MAR's buffer after the CN has received the EF-BU message from the PAR. Thus, the data packet buffering time T_{PBT} is the time required for the previous access router (PAR) to finish the transfer of packets to the new access router (NAR) [17]. T_{PBT} is represented by

$$T_{PBT} = \left| T_{\text{First-packet}} - T_{\text{polling-data-packet}} \right|, \quad (1)$$

where $T_{\text{First-packet}}$ indicates the time of the delivery of the first packet and $T_{\text{polling-data-packet}}$ defines the time for the delivery of the polling data packet through tunneling between the previous access router (PAR) going to the new access router (NAR). During T_{PBT}, only the received packets through tunneling mechanism by the previous access router (PAR) are then directed to the mobile node (MN). During this process, the waiting time, T_{PBT}, is calculated by the MAR controller until the polling data packet has arrived. It is assumed in this article that, during T_{PBT}, the modified access router (MAR) buffer size is enough for buffering packets that are directly received from the correspondent node (CN). As soon as T_{PBT} expires, the buffered packets will be delivered continuously

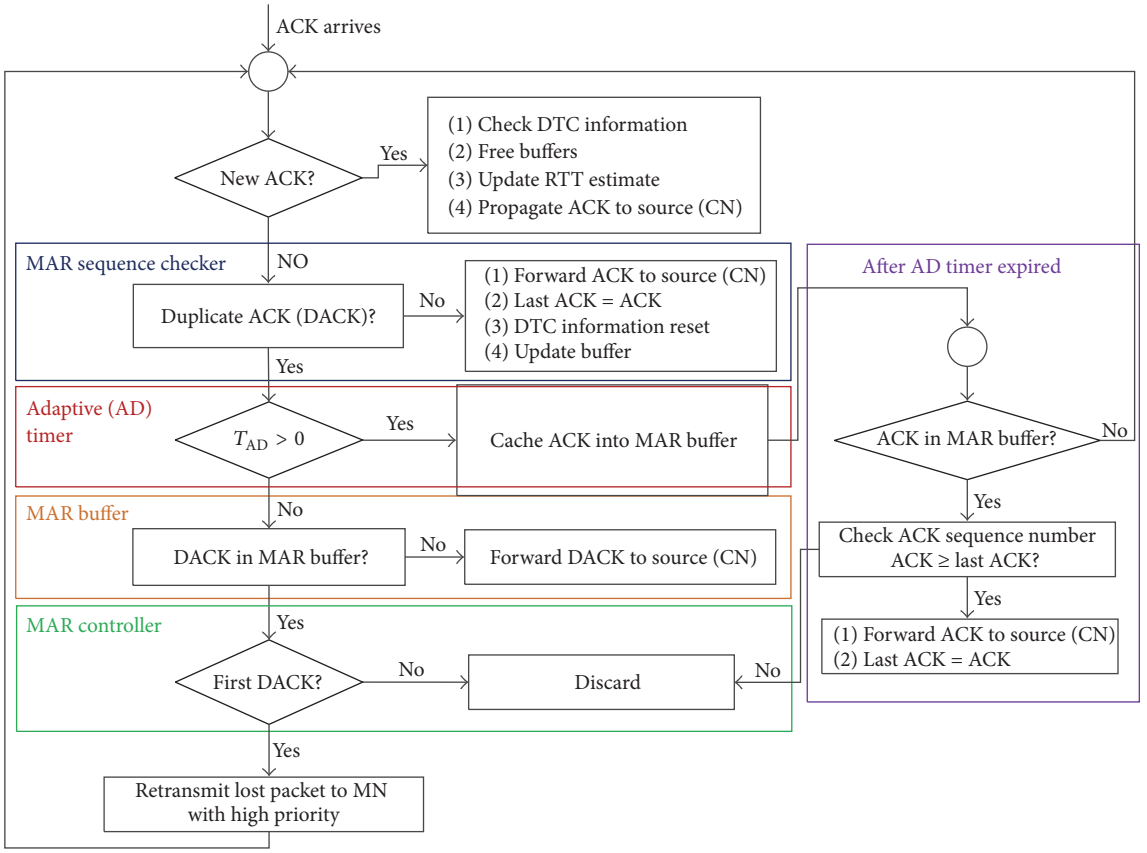


Figure 8: The flow chart of optimized MAR snoop function for ACK.

by the MAR buffer to the MN. Moreover, the NAR can periodically send control messages for notifying the buffer states to the CN or HA in order to prevent packet overflow in the MAR buffer. The CN or the MN can then control the data traffic by utilizing these messages. The NAR buffer is constructed with nonpriority First-In-First-Out.

3.3. Optimized Realignment Algorithm for ACK with Adaptive Timer. The realignment algorithm flowchart for ACK to achieve the proposed scheme is depicted in Figure 8. The MN sends ACKs whenever it received data packets from the NAR. These ACKs are processed by the MAR sequence checker in order to see whether the received ACKs are duplicates or not. The ACK is forwarded to the CN if there is no occurrence of duplication of an ACK. That is, this ACK acts as the final ACK. On the other hand, if duplication has occurred, the optimized snoop for ACK (OSA) algorithm will be processed by a snoop agent for the ACK in order to accommodate the disordered packets through delaying the ACK segment processing. An adaptive timer (AD) to delay the ACKs will be used to prevent TCP performance degradation due to DACK. The adaptive delay is denoted by T_{AD}, which is defined as the time required postponing ACKs during the schedule time. T_{AD} is derived by

$$T_{AD} = \max\left(T_{S\_PN} \;\&\; T_{OSP}\right), \tag{2}$$

where T_{S_PN} is the snoop information transmission delay between the PAR and NAR via the border gateway (BG). The time period in which disordered packets can arrive during a handover is defined by T_{OSP}. As shown in Figure 7, the packet transmission delay between a CN and a BG is denoted by T_{C_B}. T_{P_B} and T_{N_B} denote the packet transmission delay between the BG and the PAR and NAR. T_{WD} denotes the wireless transmission delay. Q_B is the queuing delay in the BG whereas Q_P and Q_N are the queuing delay in the PAR and NAR, respectively. The packet transmission delay between the PAR and NAR is denoted by T_{P_N} where tunneling is used. Thus, T_{P_N} is denoted by

$$T_{P\_N} = T_{P\_B} + T_{N\_B} + Q_B. \tag{3}$$

T_{OSP} denotes the difference between the delay times of a normal packet that is directly transmitted by the correspondent node (CN) to the new access router through the BG and a polling packet transmitted by the CN via tunneling from the PAR to the NAR via the BG. Thus, the distance between the BG and the PAR affects T_{OSP}. The polling packet transmission delay through the previous access router (PAR), that is, from the correspondent node (CN) going to the new access router (NAR), is denoted as $T_{D\text{-}Tunneling}$ in order to calculate T_{OSP}. Thus, $T_{D\text{-}Tunneling}$ is derived by

$$T_{D\text{-}Tunneling} = T_{C\_B} + Q_B + T_{P\_B} + Q_P + T_{P\_N}. \tag{4}$$

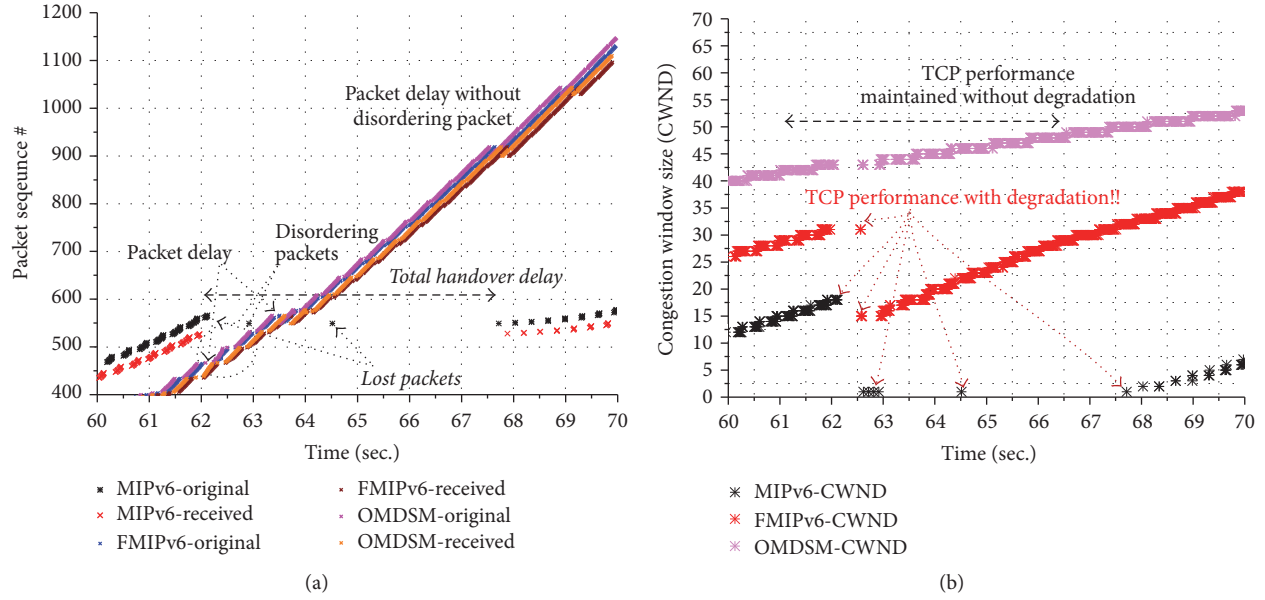


FIGURE 9: The packet transmission performance comparisons: (a) handover latency comparisons; (b) TCP performance comparisons.

Also, the packet transmission delay from the CN to the NAR directly via the BG is denoted as $T_{\text{D-Direct}}$. $T_{\text{D-Direct}}$ is represented by

$$T_{\text{D-Direct}} = T_{\text{C\_B}} + Q_{\text{B}} + T_{\text{N\_B}}. \tag{5}$$

Thus, T_{OSP} can be described as follows:

$$\begin{aligned} T_{\text{OSP}} &= T_{\text{D-Tunneling}} - T_{\text{D-Direct}} \\ &= \left(T_{\text{C\_B}} + Q_{\text{B}} + T_{\text{P\_B}} + Q_{\text{P}} + T_{\text{P\_N}}\right) \\ &\quad - \left(T_{\text{C\_B}} + Q_{\text{B}} + T_{\text{N\_B}}\right) \\ &= \left(T_{\text{P\_B}} + Q_{\text{P}} + T_{\text{P\_N}} - T_{\text{N\_B}}\right). \end{aligned} \tag{6}$$

Therefore, the duplicated ACK problem can be solved through delaying the ACKs during T_{AD}. The delayed ACKs of the content in the temporary buffer are transmitted by the MAR to the CN after T_{AD}. That is, the NAR sends the stored ACKs to the CN by arranging the ACK packets with respect to the transmission order after the NAR waits a maximum time between $T_{\text{S\_PN}}$ and T_{OSP}, if the adaptive timer has expired.

A snoop agent tries to find a duplicated ACK (DACK) in the MAR buffer if the adaptive timer has a value of less than 0. As soon as the DACK has been found, the MAR controller then determines whether it is the first DACK or not. Consequently, in order to prevent the retransmission of packets from the CN that would be detrimental to the performance of TCP in Mobile IPv6 networks, the proposed OMDSM algorithm keeps the data transmission and ACK transmission in sequence.

4. Performance Evaluation and Comparisons

In this section, the performance of the proposed scheme is evaluated using the Network Simulator (NS). The experiments are performed utilizing the simulation code that is created through the INRIA/Motorola MIPv6 code which is based on the standard Network Simulator distribution version. Two main modules have been extended with these codes: first, a data realignment algorithm and an ACK realignment algorithm that utilizes an optimized snoop protocol. It is assumed that, during the L2 handover, the DTC procedure, EF-BU, and reverse-BU procedures would be processed. That is, the DTC and BU processing time during total handover latency can be neglected. The OSDSM packet management processing requires a very short period, so, it can be ignored in the total handover latency. That is because the packet management algorithm in the router is very fast which is about 120 ηsec in the worst case [21, 22]. The original release of the code has been extended to enable it to work with two or more mobile nodes. Wired links with available bandwidths and link delays were used for the simulation network. The binding between the correspondent node and the mobile node allows the bulk data transfer, that is, by file transfer protocol. In the simulation, it is required that the size of the buffer needs to be predetermined in order to know if it is enough to cache the packets that are received directly from the correspondent node, thus, packet overflow can be prevented. The sequence number of the TCP data that are received considering its simulation time within the MIPv6, FMIPv6, and OMDSM are shown in Figure 9. It is depicted in Figure 9(a) that, during the handover between access routers, before the MN can send a binding update to the home agent (HA), some packets may have been dropped caused by route disconnection wherein it requires packet retransmission in Mobile IPv6. The congestion window size (CWND) in MIPv6 between ARs is shown in Figure 9(b). Continuous data packet losses abruptly reduce the CWND after the handover between ARs. Therefore, multiple TCP timeouts can occur if the handover latency is excessively high which causes the TCP performance degradation in MIPv6.

In FMIPv6, the figure shows the packet transmission during the handover that includes the buffering of packets between the new access router (NAR) and the previous access router (PAR). The received packets through tunneling mechanism and those that are directly sent from the correspondent node cause the disordering of packets that can be received by the mobile node even though packet loss does not happen. Thus, sending a DACK message to the CN resulted from the disordering of packets problem which leads to decrease in TCP performance. In FMIPv6's CWND between ARs, packet disruption did not occur during the handover between ARs. Nevertheless, due to the tunneling mechanism between the previous access router and the new access router, the sender is required to retransmit the packets that are delayed right after the same ACK was received three times coming from the mobile node. The CWND has been reduced by these packet retransmissions which have caused a lot of data packets to wait for a higher CWND. This is caused by the FMIPv6 handover procedure since tunneling of packets causes longer time delays leading to packet retransmissions. In OMDSM scheme, it is shown that the receiver accepts the packets normally although there is an occurrence of packet delay that is caused by traffic management in MAR; thus, the disordering of packets problem as well as packet losses can be avoided and will not require for retransmission. Therefore, the OMDSM scheme can improve the packet transmission QoS despite a slight packet transmission delay that might happen. A minor packet delay can be allowed by the OMDSM scheme which manages the data streaming in total transmission time. Thus, the congestion window size value of the sender is sustained; hence, it enhances the TCP performance. Accordingly, as compared to alternative approaches, the OMDSM scheme has achieved prominent results which can support a remarkable data streaming management without packet losses, long time packet transmission delay, and disordering of packets.

5. Conclusions

This paper introduced an optimized multimedia data streaming management algorithm in IP-based wireless/mobile networks during handover for multimedia data streaming for location-based mobile marketing applications within the Internet of Things (IoT) environment. The impact of handovers between access routers (ARs) has been analyzed for disordering of packets under the MIPv6 in a fast handover environment. The proposed OMDSM scheme shows that it can improve the TCP performance and prevent the packet disordering problem in the existing IP-based mobility management protocols. The simulation results show that the OMDSM scheme has a better performance as compared with the conventional protocols. Also, it is found out to be working satisfactorily in fast handover situations in IoT applications. In addition, a seamless multimedia streaming can be supported with the right sequence of packets with the integration of the OMDSM scheme for location-based mobile marketing applications in an IoT environment.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT, and future Planning (2015R1A2A2A03002851).

References

[1] V. Tsaotissidis, "Open issues on TCP for mobile computing," *Journal of Wireless Communication and Mobile Computing*, vol. 1, no. 2, 2002.

[2] S. J. Vaughan-Nichols, "Mobile IPv6 and the future of wireless Internet access," *IEEE Computer*, vol. 36, no. 2, pp. 18–20, 2003.

[3] G. Al-Gadi, A. Babiker, and N. Mustafa, "Comparison between IPv4 and IPv6 using OPNET simulator," *IOSR Journal of Engineering*, vol. 4, no. 8, pp. 44–50, 2014.

[4] D. B. Johnson, C. E. Perkins, and J. Arkko, "Mobility support in IPv6," RFC 3775, IETF, 2004.

[5] A. J. Jabir, S. Shamala, Z. Zuriati, and N. Hamid, "A comprehensive survey of the current trends and extensions for the proxy mobile IPv6 protocol," *IEEE Systems Journal*, pp. 1–17, 2015.

[6] A. Moravejosharieh and H. Modares, "A proxy MIPv6 handover scheme for vehicular ad-hoc networks," *Wireless Personal Communications*, vol. 75, no. 1, pp. 609–626, 2014.

[7] F. Li, X. Wang, T. Pan, and J. Yang, "Packet delay, loss and reordering in IPv6 world: a case study," in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC '16)*, pp. 1–6, February 2016.

[8] R. Koodli, "Fast handovers for mobile IPv6," RFC 4068, 2005.

[9] D. Lee, C. Oh, S. Lee, J. Park, and K. Kim, "Design and analysis of the mobile agent preventing out-of-sequence," in *Proceedings of the International Conference on Information Networking (ICOIN '99)*, Tokyo, Japan, January 1999.

[10] D. S. Eom, M. Sugano, M. Murata, and H. Miyahara, "Performance improvement by packet buffering in mobile IP based networks," *IEICE Transactions on Communications*, vol. 83, no. 11, pp. 2501–2512, 2000.

[11] D. Lee, G. Hwang, and O. Changhwan, "Performance enhancement of mobile IP by reducing out-of-sequence packets using priority scheduling," in *Proceedings of the APCC*, pp. 513–516, November 2001.

[12] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 756–769, 1997.

[13] J. Gu, "The adaptive header compression algorithm of mobile IPv6 Network," *Lecture Notes in Electrical Engineering*, vol. 238, pp. 1603–1610, 2014.

[14] A. K. Barbudhe, V. K. Barbudhe, and C. Dhawale, "Comparative analysis of security mechanism of mobile IPv6 threats against binding update, Route Optimization and Tunneling," in *Proceedings of the 6th IEEE International Conference on Adaptive Science and Technology (ICAST '14)*, 7, 1 pages, October 2014.

[15] A. Dhamdhere, M. Luckie, B. Huffaker, K. Claffy, A. Elmokashfi, and E. Aben, "Measuring the deployment of IPv6: topology, routing and performance," in *Proceedings of the ACM Internet Measurement Conference (IMC '12)*, pp. 537–550, Boston, Mass, USA, November 2012.

[16] B. J. Park, H. In, and H. A. Latchman, "An approach to efficient and reliable media streaming scheme," in *Proceedings*

of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (IEEE ISBMSB '06), April 2006.

[17] B. J. Park and H. A. Latchman, "Performance enhancement of fast handover for MIPv6 by reducing out-of-sequence packets," *Wireless Personal Communications*, vol. 47, no. 2, pp. 207–217, 2008.

[18] A. K. Quoc, D. S. Kim, and H. Choo, "A novel scheme for preventing out-of-order packets in fast handover for Proxy Mobile IPv6," in *Proceedings of the 28th International Conference on Information Networking (ICOIN '14)*, pp. 422–427, February 2014.

[19] N. Kwon, H. Kim, S. Oh, and H. Choo, "Fast handover scheme based on mobility management of head MAG in PMIPv6," in *Computational Science and Its Applications—ICCSA 2011*, vol. 6786 of *Lecture Notes in Computer Science*, pp. 181–193, Springer, 2011.

[20] I. Al-Surmi, M. Othman, N. A. W. A. Hamid, and B. M. Ali, "Latency low handover mechanism considering data traffic lost preventing for proxy mobile IPv6 over WLAN," *Wireless Personal Communications*, vol. 70, no. 1, pp. 459–499, 2013.

[21] V. Srinivasan and G. Varghese, "Fast address lookups using controlled prefix expansion," *ACM Transactions on Computer Systems*, vol. 17, no. 1, pp. 1–40, 1999.

[22] R. Kawabe, S. Ata, M. Murata, M. Uga, K. Shiomoto, and N. Yamanaka, "On performance prediction of address lookup algorithms of IP routers through simulation and analysis techniques," in *Proceedings of the International Conference on Communications (ICC '02)*, pp. 2146–2151, May 2002.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 6389567, 17 pages
https://doi.org/10.1155/2017/6389567

Research Article

Detecting Difference between Process Models Based on the Refined Process Structure Tree

Jing Fan, Jiaxing Wang, Weishi An, Bin Cao, and Tianyang Dong

College of Computer Science and Software Engineering, Zhejiang University of Technology, Hangzhou 310023, China

Correspondence should be addressed to Bin Cao; bincao@zjut.edu.cn

Received 20 January 2017; Accepted 21 February 2017; Published 15 March 2017

Academic Editor: Jaegeol Yim

The development of mobile workflow management systems (mWfMS) leads to large number of business process models. In the meantime, the location restriction embedded in mWfMS may result in different process models for a single business process. In order to help users quickly locate the difference and rebuild the process model, detecting the difference between different process models is needed. Existing detection methods either provide a dissimilarity value to represent the difference or use predefined difference template to generate the result, which cannot reflect the entire composition of the difference. Hence, in this paper, we present a new approach to solve this problem. Firstly, we parse the process models to their corresponding refined process structure trees (PSTs), that is, decomposing a process model into a hierarchy of subprocess models. Then we design a method to convert the PST to its corresponding task based process structure tree (TPST). As a consequence, the problem of detecting difference between two process models is transformed to detect difference between their corresponding TPSTs. Finally, we obtain the difference between two TPSTs based on the divide and conquer strategy, where the difference is described by an edit script and we make the cost of the edit script close to minimum. The extensive experimental evaluation shows that our method can meet the real requirements in terms of precision and efficiency.

1. Introduction

A business process is a series of activities to reach a certain goal, such as approval for vacation, purchase order, or claims for travel expense. It is a workflow if a business process is automated by a supporting software system. Workflow management systems (WfMS) are used to define, execute, and monitor the workflows [1].

Advances in wireless network technology and the widespread use of hand-held terminals enable the realization of mobile workflow management systems (mWfMS), such as Exotica/FMDC [2] and WHAM [3]. A workflow is called mobile if it contains activities that are performed by actors with a mobile device (e.g., mobile phone or PDA). The typical users in mobile workflows are travelling salesman, service technicians, or maintenance engineers [4]. And the mWfMS sometimes have location constraints, which means the location of user should be also considered by mWfMS when allocating activities [5], that is, allocating an activity that has to be performed to the actor with the shortest travel path or at a certain location. Thus, it is necessary for workflow system to know about the current location of mobile users [6].

The development of mWfMS leads to large number of business process models, which are valuable assets. However, different locations for the same business process may result in different execution orders of activities. For example, one company has two offices: Hangzhou and Beijing, and Beijing is the headquarters. For some businesses of Hangzhou, the corresponding materials need to be sent to Beijing. After approval with signature, the materials will be sent back to Hangzhou; in this way, this business process can be successfully executed, while for the same business process of Beijing, there is no need to send the materials. Determining these differences is so meaningful that we can find out the reason of inefficiency during the execution of the business process. That is, detecting difference between process models is helpful for users to quickly locate the difference and rebuild the process model.

Graph edit distance (GED) [7] is a good way to measure the similarity (or dissimilarity) between graphs. Process

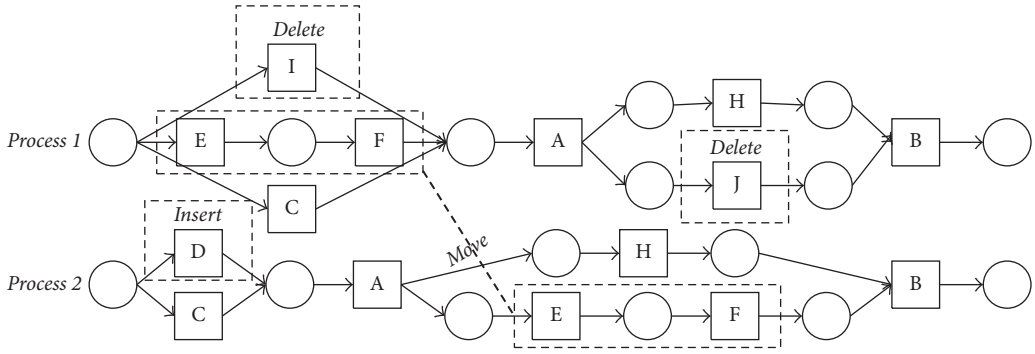


Figure 1: Two Petri net modeled process models.

model is generally represented as a graph, while GED cannot be directly used to compute the difference between two process models, since GED is applicable for the graphs that only contain one type of node. Thus, it is not suitable for measuring the dissimilarity between the graphs with more than two kinds of nodes, such as Petri net based process model.

Vanhatalo et al. provide a feasible model to detect difference between process models. They parse a workflow graph to its corresponding refined process structure tree (PST), that is, decomposing a workflow graph into a hierarchy of subworkflows that are subgraphs with a single entry and a single exit [8]. But this model cannot be directly used; it is because a leaf node of PST represents an edge of its corresponding process model. In order to process the task nodes, that is, mapping the task nodes or generating the node based edit operations (such as node delete or node insert), we need to parse the edge of PST and get the task nodes. To conveniently obtain and process the nodes, we present a method to parse the PST to its corresponding task based process structure tree (TPST) by referencing the work of Cao et al. [9], where a leaf node of TPST is a task node and a nonleaf node represents a control flow structure of its corresponding process model. Therefore, the problem of detecting difference between process models is transformed to detect difference between their corresponding TPSTs.

Zhang et al. show that the problem of computing the edit distance between labeled trees is NP-complete [10]. In order to efficiently compute the difference between two TPSTs, we present an algorithm that uses the divide and conquer strategy to generate an edit script that we try to make its cost close to minimum. There are three steps to reach this goal: (1) two TPSTs that correspond with two process models are split into several fragments, and the mapped fragment pairs of two TPSTs are found; (2) the mapped nodes in each mapped fragment pair are determined; (3) the edit script of two TPSTs is generated based on the mapped nodes and fragment pairs. In this paper, the process models are modeled by Petri net [11].

Generally, we consider the difference as an edit script that consists of a set of edit operations. In this paper, we consider three kinds of edit operations: *node delete*, *node insert,* and *fragment move. Node delete* and *node insert* are the basic edit operations, which mean deleting and inserting a node; they are complementary. The reason why we consider *fragment move* is that it can be represented by a set of node deletes and inserts, while the move of fragment can be more understandable. For example, there are two process models that are modeled by Petri net in Figure 1: *Process 1* and *Process 2*. The following edit script can transform Process 1 to Process 2: deleting nodes I and J, inserting node D, and moving a fragment that consists of E and F in *Process 1* to the position where the same fragment in *Process 2* is.

The contributions of this paper are highlighted as follows:

(1) The implementation of parsing PST to TPST is performed in this paper, and we transformed the problem of computing difference between two process models into computing difference between two TPSTs.

(2) The divide and conquer strategy is used to determine the mapped fragment pairs of two TPSTs and then the mapped nodes are determined in each mapped fragment pair, which can narrow the range of node mapping and improve the efficiency of difference detecting.

(3) We design an algorithm to generate an edit script between two TPSTs, where we make the cost of this edit script close to minimum.

(4) On the basis of the real data, we conduct extensive experiments to evaluate the performance of our algorithm in terms of precision and execution time.

The rest of this paper is organized as follows. The preliminaries are described in Section 2. Section 3 presents the method of parsing the PST to its corresponding TPST. Section 4 introduces the difference detection algorithm. The evaluation in terms of precision and efficiency is performed in Section 5. Section 6 introduces the related work and Section 7 concludes this paper.

2. Preliminaries

In this section, some preliminaries are introduced. Sections 2.1 and 2.2 introduce refined process structure tree (PST) and task based process structure tree (TPST), respectively. Section 2.3 presents some basic notions that are used in our algorithm. The basic edit operations are defined in Section 2.4, and edit script is described in Section 2.5. An edit script consists of a series of edit operations, which can

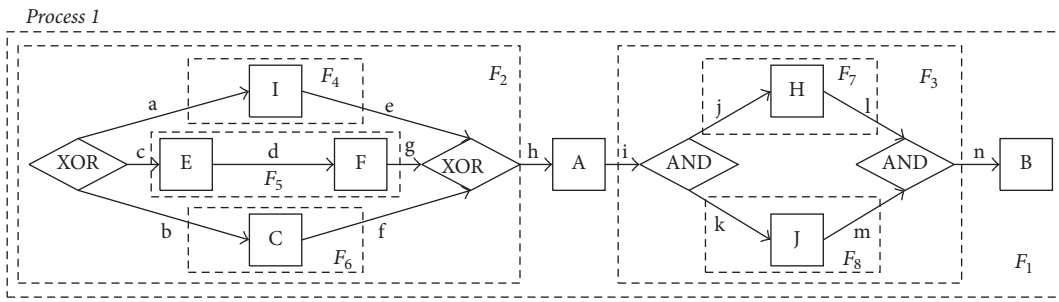


Figure 2: Blocks of a process model.

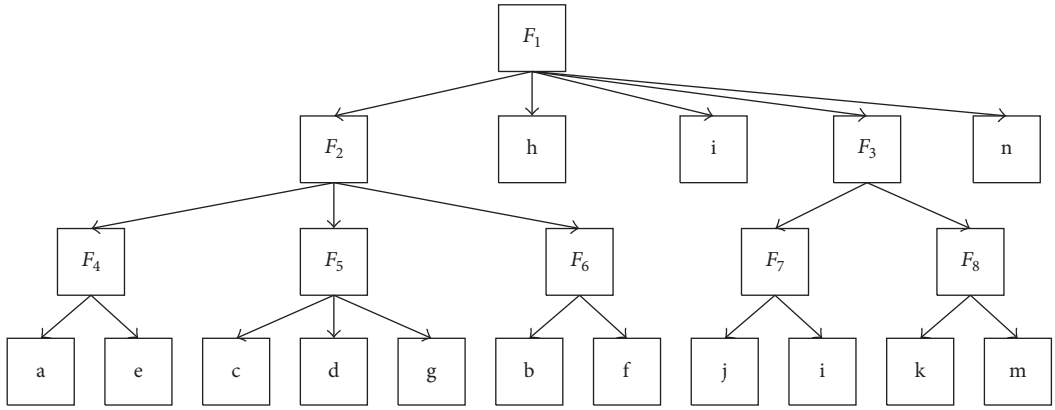


Figure 3: Refined process structure tree.

transform one TPST to the other. Section 2.6 introduces the costs of edit operations and edit script.

2.1. Refined Process Structure Tree (PST)

Definition 1 (blocks of a process model). A block of a process model is a nonempty submodel, which is defined as a quadruple $B = (Entry, Exit, V, E)$, where *Entry* and *Exit* are the single entry node and the single exit node of the block, respectively, V is the node set, and E is the edge set.

For example, in Figure 2, *Process 1* is a process model and it is decomposed into 8 blocks: $F_1 \sim F_8$. The whole process model is block F_1, where the leftmost route node *XOR* is its entry node and the rightmost task node B is its exit node. Thus, *Process 1* can be represented by these nested and nonoverlapping blocks: *Process 1* = $\{F_1(F_2(F_4, F_5, F_6), F_3(F_7, F_8))\}$, where F_1 contains F_2 and F_3, the nested blocks of F_2 are F_4, F_5, and F_6, and F_3 includes F_7 and F_8.

Generally, there are four kinds of control flow structures in a process model: *Sequence*, *Exclusive*, *Parallel,* and *Loop*. It is hard to compute the difference between two process models since a process model has several structures and these different structures can be assembled in an arbitrary way. Parsing a process model to its corresponding tree model is a good way to simplify this problem. It is because a tree structure is simpler than the structure of a process model and we can easily obtain all nodes and their relationships in a tree. Thus, we use the existing tree model called the refined process structure tree (PST) to represent a process model. That is, a process model is decomposed into several blocks that consists of a single entry node and a single exit node [8], and these blocks are organized in a hierarchy way. The blocks are organized into a PST in a hierarchy way, and these blocks represent the route nodes of PST and the leaf nodes of PST corresponds with the edges of its corresponding edges.

For example, the process model *Process 1* in Figure 2 can be parsed to its corresponding PST in Figure 3. The route node F_5 in PST represents the block named F_5 in the process model. The leaf nodes of F_5 in PST are c, d, g, which correspond with three edges in the process model: c, d, and g, respectively. PST can represent the hierarchy relationships of blocks, but its corresponding leaf nodes are unordered. Therefore, it cannot represent the control flow structures of a process model well since some structures are ordered while some structures are unordered. That is, we need a semiordered tree model to represent a process model. Thus, we reference the work of Cao et al. [9]. and improve PST to a new semiordered tree model called task based process structure tree (TPST).

2.2. Task Based Process Structure Tree (TPST).

The differences between TPST and PST are listed as follows:

(1) A leaf node of a TPST is a task node of its corresponding process model.

(2) A nonleaf node, that is, a route node, can only be labeled as "*Sequence,*" "*Loop,*" "*XOR,*" or "*AND,*" where "*XOR*" and "*AND*" represent the exclusive and parallel structures, respectively.

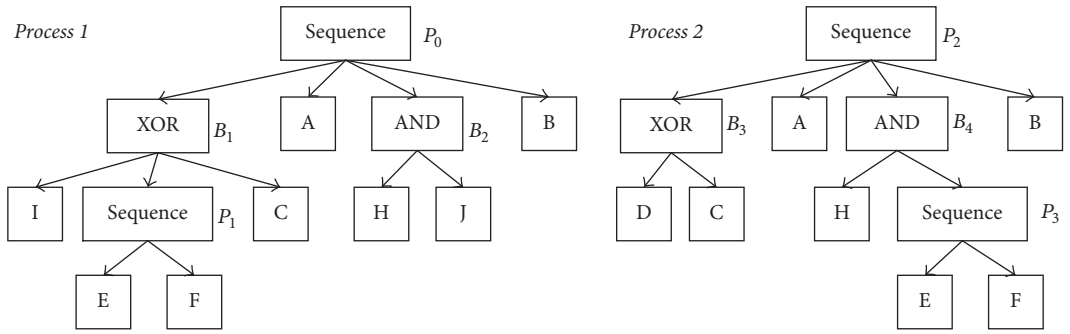


Figure 4: Task based process structure tree.

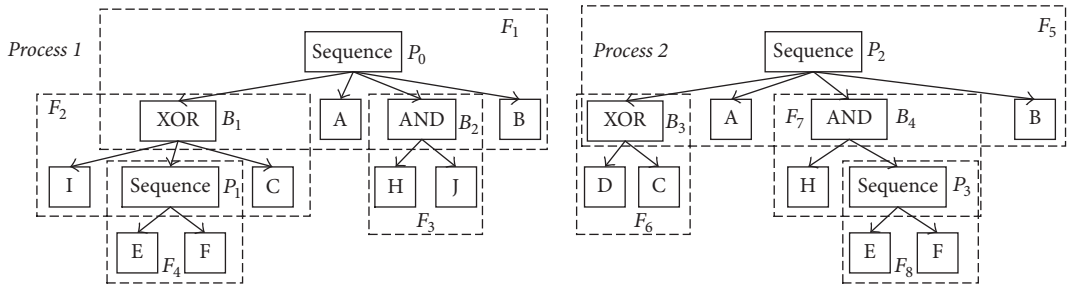


Figure 5: Fragments of two TPSTs.

(3) TPST is a semiordered tree: if a nonleaf node is labeled as "*Sequence*" or "*Loop,*" its child nodes are ordered; otherwise, its child nodes are unordered.

Thus, a TPST can describe a process model that contains ordered and unordered control flow structures. As shown in Figure 4, there are two TPSTs that are parsed from *Process 1* and *Process 2* of Figure 1. The leaf nodes are the task nodes of the process models and the nonleaf nodes are the route nodes that represent the control flow structures. The labels of route nodes are marked beside the route nodes: for example, the label of the root node "*Sequence*" in *Process 1* is "P_0."

2.3. Basic Notions

Definition 2 (fragment of a TPST). A fragment of a TPST consists of a route node and its adjacent child nodes. It is a tuple $F = (root, node, type)$, where

(1) *root* is the root node of this fragment, which is usually a route node. *Root* has two kinds of type: (1) ordered, where child nodes of *root* form a sequence, and (2) unordered, where child nodes of *root* form a set with no order,

(2) *node* is the node set of this fragment that contains *root* and its child nodes that are directly connected to *root*,

(3) *type* is the type of this fragment that is the same as the type of its root node.

For example, *Process 1* of Figure 5 can be split into 4 fragments: F_1(*Sequence*, {*Sequence*, *XOR*, *A*, *AND*, *B*}, *ordered*), F_2(*XOR*,{*XOR*, *I*, *Sequence*, *C*}, *unordered*), F_3(*AND*, {*AND*, *H*, *J*}, *unordered*), and F_4(*Sequence*, {*Sequence*, *E*, *F*}, *ordered*) and there are also 4 fragments in *Process 2*: F_5(*Sequence*,{*Sequence*, *XOR*, *A*,*AND*,*B*}, *ordered*), F_6(*XOR*, {*XOR*, *D*,*C*}, *unordered*), F_7(*AND*, {*AND*, *H*, *Sequence*}, *unordered)*, and F_8(*Sequence*, {*Sequence*, *E*, *F*}, *ordered*).

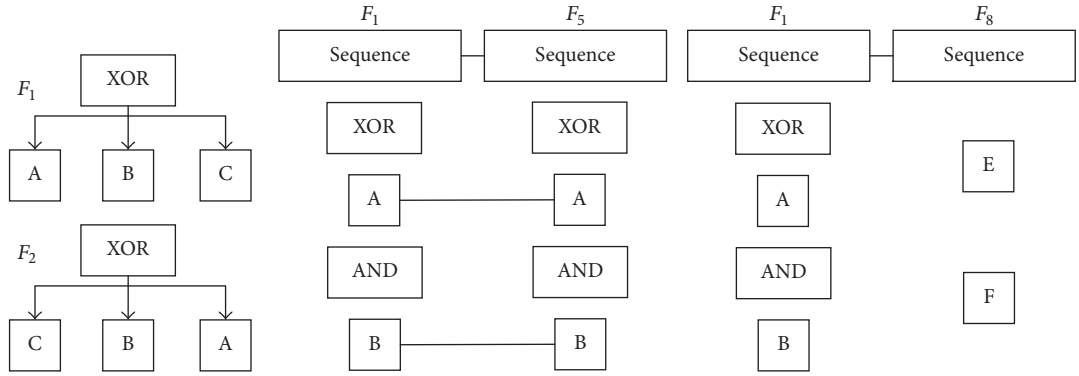
Definition 3 (node mapping). There are two kinds of nodes in a TPST: task node and route node. Two task nodes can be mapped if their labels are identical. Since a route node is the root node of a fragment, whether two route nodes can be mapped depends on the similarity of their corresponding fragments. The more similar the two fragments are, the more possible the two route nodes can be mapped.

Definition 4 (similarity score of two fragments). Let F_1 and F_2 be two fragments that are from two different TPSTs; the similarity score of F_1 and F_2 is the ratio of their mapped nodes to total nodes, which can be computed according to the following equation:

$$\mathrm{Sim}(F_1, F_2) = \frac{2 \times |MapNodes|}{|node_1| + |node_2|}. \quad (1)$$

$|MapNodes|$ is the number of mapped nodes in two fragments: F_1 and F_2. For two unordered fragments, their mapped nodes are the intersection set of their node set. For two ordered fragments, the nodes in their node sequence (Definition 5) meet the longest common node subsequence which are the mapped nodes. $|node_1|$ and $|node_2|$ are the node set size of F_1 and F_2, respectively.

Definition 5 (node sequence). Let F_1 be a fragment, and let SN_1 be its sequence of node labels that consists of the label of



(a) Two exclusive fragments (b) Mapping result

Figure 6: Ordered fragment mapping.

root node and the task nodes' labels from left to right. In the node sequence, we do not consider the route nodes that are leaf nodes.

For example, in Figure 6, the ordered fragments in *Process 1* are $\{F_1, F_4\}$, and $\{F_5, F_8\}$ are ordered fragments of *Process 2*. The node sequences of F_1 and F_5 are $\mathrm{SN}_1 = \{\text{Sequence}, A, B\}$ and $\mathrm{SN}_5 = \{\text{Sequence}, A, B\}$, respectively.

Before introducing the longest common node subsequence (LCNS), we describe the related notations: subsequence, common subsequence, and longest common subsequence [12].

Definition 6 (subsequence). Let $X = (x_1, x_2, \ldots, x_m)$ be a sequence; if there exists $1 \leqslant i_1 < i_2 < \cdots < i_k < m$, s.t. $Z = (z_1, z_2, \ldots, z_k) = (x_{i1}, x_{i2}, \ldots, x_{ik})$, then Z is regarded as the subsequence of X, which is marked as $Z < X$.

For example, $X = (A, B, C, B, D, A, B)$ and $Z = (B, C, B, A)$, $Z < X$.

Definition 7 (common subsequence). Let X and Y be two sequences, and $Z < X$, $Z < Y$; then Z is the common subsequence of X and Y.

Definition 8 (longest common subsequence, LCS). Let X and Y be two sequences, and $Z < X$, $Z < Y$. Z is the longest common sequence of X and Y iff $\forall$ *subsequence* Z' of X and Y, $|Z| \geq |Z'|$.

For example, for two ordered fragments F_1 and F_5 in Figure 5, the longest common node subsequence of their corresponding node sequences S_1 and S_5 is LCNS $(S_1, S_5) = \{Sequence, XOR, A, AND, B\}$. In order to obtain the longest common node subsequence, the dynamic programming method is applied [13].

2.4. Edit Operations. In this paper, we use three kinds of edit operations to describe the difference between two TPSTs: *node delete*, *node insert*, and *fragment move*.

Node Delete: Delete(x). Node x can be directly deleted if it is the leaf node; otherwise, before deleting node x, node x's child nodes are connected to node x's parent node.

For example, as shown in Figure 5, the leaf node A of *Process 1* can be directly deleted: $Delete(A)$. Before deleting node B_1, its child nodes $\{I, P_1, C\}$ are connected to its parent P_0.

Node Insert: Insert$(x, a, position)$. Node x is inserted as the child node of a. If node a is unordered, the default *position* is 0 and x can be inserted in an arbitrary position. Otherwise, x is inserted as a's *position*-th child node.

For example, as shown in Figure 5, to insert node D of *Process 2*, the first step is to determine the parent node that node D is going to be inserted in *Process 1*, that is, node B_1. Since B_1 is unordered, D can be inserted as B_1's child node in an arbitrary position, that is, $Insert(D, B_1, 0)$.

Fragment Move: Move$(f, a, position)$. A fragment f is moved as the *position*-th child fragment of node a. The *position* is 0 if a is a unordered route node, which means that f can be inserted as a's child fragment in an arbitrary position.

For example, in Figure 5, F_4 and F_8 are identical but they connect to different parent nodes: B_1 and B_4, respectively. Since B_1 and B_4 are not mapped nodes, F_4 will be removed. The mapped node of B_4 is B_2, so F_4 is removed as the child fragment of B_2 in an arbitrary position since B_2 is unordered, that is, $Move(F_4, B_2, 0)$.

2.5. Edit Script. In this section, $TPST_1$ refers to the TPST that the edit operations are applied, and $TPST_2$ is the resulting TPST. Formally, suppose that e is an edit operation, a sequence E that consists of a set of operations, $E = \{e_1, e_2, \ldots, e_m\}$, can transform $TPST_1$ into $TPST_2$, which is denoted by $TPST_1 \rightarrow_E TPST_2$. We call such a sequence an edit script of transforming $TPST_1$ to $TPST_2$, which is also the difference of $TPST_1$ and $TPST_2$.

For example, in Figure 5, the edit script of transforming *Process 1* into *Process 2* is $editScript(Process\ 1 \rightarrow Process\ 2) = \{Delete(I), Delete(J), Insert(D, B_1, 0), Move(F_4, B_2, 0)\}$.

2.6. Cost Model. There exist a large number of edit scripts that can transform one TPST to another TPST. For example, in Figure 5, two kinds of edit scripts can be applied to transform *Process 1* into *Process 2*: (1) $Process\ 1 \rightarrow_{E1} Process\ 2 = \{Delete(I), Delete(J), Insert(D, B_1, 0), Move(F_4, B_2, 0)\}$. (2) $Process\ 1 \rightarrow_{E2} Process\ 2 = \{Delete(I), Delete(E), Delete(F),$

$Delete(P_0), Delete(J), Insert(D, B_1, 0), Insert(P_3, B_2, 0), Insert(E, P_3, 0), Insert(F, P_3, 0)\}$ For these two edit scripts E_1 and E_2, which one is better? Or there are thousands of edit scripts that can be applied to convert one TPST to another, which edit script is the best? In order to solve this problem, the cost of edit script is proposed to evaluate whether an edit script is good or not. The smaller the cost is, the better the edit script is.

In this paper, we adopt a simple cost model that the cost of each edit operation is equal to 1; for example, $Cost_{Delete}(x) = Cost_{Insert}(x) = Cost_{Move}(f) = 1$, which represents that deleting a node x, inserting a node x, and moving a fragment f, respectively, have a unit cost.

Then the cost of an edit script E is the sum of all the costs of its corresponding edit operations $e_1, e_2, \ldots, e_m$; that is, $C(E) = C(e_1) + C(e_2) + \cdots + C(e_m)$. For example, as mentioned above, $C(E_1) = 4$ and $C(E_2) = 9$, so the edit script E_1 is better than E_2.

3. Parsing PST to TPST

There exists a method to parse a process model to its corresponding refined process structure tree (PST); that is, a process model is decomposed into a hierarchy of subprocess models with a single entry and a single exit. However, it is inconvenient to compute difference between process models by their corresponding PSTs. It is because a leaf node of PST represents an edge of its corresponding process model. The difference between two process models is described by using the edit operations that include node operations and fragment operation. In order to describe the node operations, we need to further parse out the task nodes through the edges. For convenience, we parse the task nodes from PST in advance and arrange the task nodes and route nodes to form a task based process structure tree (TPST).

TPST is a task based process description, where leaf nodes are task nodes and nonleaf nodes represent control flow structures. The edit operations can be described in the TPST more intuitively by comparing with PST; for example, we can observe which nodes are inserted or deleted or which control flow structures are moved. Besides, TPST is more convenient to design difference detection algorithm because we need to frequently process the task nodes.

Main Idea. The main idea of parsing PST to TPST is parsing the task nodes and route nodes from PST; then all parsed nodes are arranged in a tree, where the structure of TPST is the same as the PST. In terms of implementation, there are three phases to parse a PST to its corresponding TPST: (1) parsing a node of PST to a node of TPST: each node in a PST is converted to be a TPST node, where a leaf node of TPST is a task node and a nonleaf node represents a control flow structure; (2) constructing the TPST: all nodes in a TPST are organized into a tree according to the hierarchy structure of PST; and (3) checking the TPST: the route node needs to be deleted if its type is "Sequence" and it has only one child node. The purpose of this phase is to better understand TPST since it is not necessary to describe a single task node by using a "Sequence."

```
Input: PST pst
Output: the root node of TPST tpst
(1) Map < PSTNode, TPSTNode > map = ∅;
(2) for each node p of pst in level-order do
(3)     TPSTNode t = transToTPSTNode(p);
(4)     map.put(p, t);
(5) end
(6) map.put(pst.exit, transToTPSTNode(pst.exit));
(7) root_tpst = constructTPST(pst, map);
(8) root_tpst = checkTPST(tpst);
(9) return root_tpst;
```

Algorithm 1: Converting PST to TPST.

Algorithm. Algorithm 1 gives the overall pseudo code for parsing PST to TPST, where the input is a PST that corresponds with a process model, and the output is the root node of its corresponding TPST. Firstly, a map named *map* that records the mapping relationship between PST nodes and TPST nodes is initialized; that is, we can obtain the relationship as to which TPST node is transformed by which PST node from this map (line (1)). Then each node in PST is iterated and transformed to a TPST node by using the function *transToTPSTNode* and the mapped node pair: PST node and TPST node are saved into *map* (line (2)–line (5)). Since we parse the task nodes by obtaining the entry nodes of the block in PST and the exit node of the total process model has not been parsed, we parse this exit node in the end (line (6)). Then, a TPST is constructed based on the parsed TPST nodes, where the hierarchy structure is arranged by referencing its corresponding PST. This phase is implemented by the function: *constructTPST*, which outputs the root node of TPST (line (6)-line (7)). In order to make the TPST more understandable, we delete the "Sequence" node that directly connects with its single task node, which is operated by the function: *checkTPST* (line (8)).

3.1. Phase 1: Parsing a Node of PST to a Node of TPST

Main Idea. There are two types of nodes in a PST: leaf node and route node. A leaf node represents an edge, and a route node is a block that contains a set of edges of the original process model. To parse a PST node to its corresponding TPST node, we need to handle different types of nodes in different ways. For a leaf node of PST, we obtain the entry node of its corresponding block and we save it if this entry node is a task node; otherwise, the entry node is discarded. For the route nodes in PST, we just save it.

Algorithm. Algorithm 2 gives the pseudo code for the first phase, where the input is a node of PST and the output is the corresponding TPST node. Firstly, if this PST node is a leaf node, we get the entry node of its corresponding edge (line (1)-line (2)). There are two possibilities of the entry node: route node or task node. The entry node will be abandoned if it is a route node since we just need the task node (line (3)-line (4)). While if the entry node is a task node, we save it as a TPST node by copying its type and label (line (5)–line (9)).

```
Input: PSTNode p
Output: TPSTNode t
(1) TPSTNode t = new TPSTNode();
(2) if p is a leaf node then
(3)   if Entry(p) is a route node then
(4)         return null;
(5)   else
(6)         t.type = "Active";
(7)         t.label = Entry(p).getName();
(8)         return t;
(9)   end
(10) else
(11) t.label = getName();
(12) t.type = getNodeType(p);
(13) return t;
(14) end
```

Algorithm 2: transToTPSTNode.

Then, if this PST node is a route node, it is saved by copying its name and type, where there are two types: ordered and unordered (line (10)–line (14)).

Example. Figure 3 is a PST of *Process 1* in Figure 1; we can observe the parsing result of this phase in Figure 4. The leaf nodes, a, b, c, h, j, k, and n, are abandoned since the entry nodes of their corresponding edges are route nodes. While the rest leaf nodes, d, e, f, g, i, l, and m, are remained because their corresponding entry nodes, E, I, C, F, A, H and J, B, are task nodes that correspond with the leaf nodes in the TPST. The nonleaf nodes, F_1–F_8 in PST, are unchanged, which correspond with the route nodes $P_0, B_1, B_2, P_2, P_1, P_3, P_4, P_5$, respectively.

3.2. Phase 2: Constructing the TPST

Main Idea. The overall structure between PST and TPST is the same; that is, their corresponding process model is decomposed into the same subprocess models and these subprocess models are organized into the same way. The difference between them is that PST is an edge based process and TPST is a task based process; that is, from the route nodes of PST, we can observe which edges a block contains and we can observe that which task nodes a fragment has from a route node of TPST. However, the organization way between route nodes in PST is the same as in TPST. So after obtain all nodes of TPST, we construct the TPST by referencing the structure of PST.

Algorithm. Algorithm 3 gives the pseudo code for the second phase, where the input is the PST and the map that records the mapping relationship between PST nodes and TPST nodes, and the output is the root node of the TPST. Firstly, for each route node p in PST, its corresponding route node t of TPST is found (line (1)–line (3)). If p is a root node then t is also a root node (line (4)–line (6)). Then, the child nodes of p are obtained, which need to be ordered according to the original process model if the type of t is ordered (line (9)–line (12)). It

```
Input: PST pst, Mao < PSTNode, TPSTNode > map
Output: TPSTNode root_tpst
(1) for each route node p of pst in level-order do
(2)   if Children(p).size is not 0 then
(3)     TPSTNode t = map.get(p);
(4)     if p is the root node of pst then
(5)       root_tpst = t;
(6)     end
(7)     List<PSTNode> child_pst;
(8)     List<TPSTNode> child_tpst;
(9)     child_pst = Children(p);
(10)    if Type(t) == Sequence or Loop then
(11)        Sort(child_pst);
(12)    end
(13)    for each node c of child pst do
(14)      map.get(c).setParent(t);
(15)      child_tpst.add(map.get(c));
(16)    end
(17)    t.setChild(child_tpst);
(18)  end
(19) end
(20) return root_tpst;
```

Algorithm 3: Construct TPST.

is because PST cannot reflect the control flow structures that are ordered and unordered. To construct the ordered control flow structures in TPST, we first need to rank the ordered blocks in PST, and then we construct the ordered structure in TPST by referencing that in PST. According to the parent-child relationships in PST, that is, a node has which child nodes and which node is the parent node of this node, the TPST also has this kind of relationships (line (13)–line (18)). After all node relationships in TPST have been constructed, the root node is returned (line (20)).

Example. The original process model is *Process 1* in Figure 1, Figure 3 is its corresponding PST, and Figure 4 is the resulting TPST by Phase 2. We can observe that the organization way between route nodes in PST is the same as TPST.

3.3. Checking the TPST

Main Idea. The main idea of this phase is to optimize the structure of TPST and better understand it. In the TPST that we obtain from the first two phases, there exist many sequence route nodes with only one task node, which need to be deleted. The reasons why we delete them are listed as follows. (1) Generally, the sequential relationship is used to describe the relationship between more than two nodes, and it is not suitable for a single node. (2) Once the task node is to be deleted, its corresponding sequence route node is also to be deleted, which leads to more edit operations.

Algorithm. Algorithm 4 gives the pseudo code for this phase, where the input is the root node of TPST, and its new root node is output. For each route node in TPST, if its type is "Sequence" and it has only one child then it needs to be deleted (line (1)-line (2)). Firstly, we get the child node and

```
Input: TPSTNode root_tpst
Output: TPSTNode root_tpst
(1) for each route node g in level-order by visiting root_tpst do
(2)   if Children(g).size == 1 and type(g) == Sequence then
(3)     children = Children(g);
(4)     parent = Parent(g);
(5)     parent.child.remove(g);
(6)     parent.child.add(children);
(7)   end
(8) end
(9) return root_tpst;
```

Algorithm 4: Check TPST.

```
Input: TPST t1, TPST t2
Output: The edit script: editScript
(1) List<Fragment> M_F = Fragment_Mapping(t1, t2);
(2) List<Node> M_N = Node_Mapping(M_F);
(3) EditScript editScript = EditScript_Generation(t1, t2, M_F, M_N);
```

Algorithm 5: Overall algorithm.

parent node of this route node (line (3)-line (4)). Then, this route node is deleted; that is, the only child node of this route node is directly connected with its parent node (line (5)-line (6)).

Example. The parsing result of Phases 1 and 2 is shown in Figure 2, and after Phase 3 we obtain the TPST that is shown in Figure 4. We can observe that all sequence route nodes with only one task node are deleted, for example, P_2, P_3, P_4, and P_5, and their parent nodes are directly pointed to their single task nodes, respectively. For example, in Figure 4, P_2 is the route node that needs to be deleted, B_1 is its parent node, and its child node is I. After P_2 is deleted, its parent node B_1 is the parent node of I.

4. Difference Detection

In this paper, the difference between two process models is described by using the node operations and fragment operation, which is shown in Section 2.4. Thus, TPST can reflect the difference in an understandable way, that is, which nodes and fragments are changed. Besides, TPST is more convenient for us to design the difference detection algorithm. Therefore, we transform the problem of detecting difference between process models into detecting difference between their corresponding TPSTs.

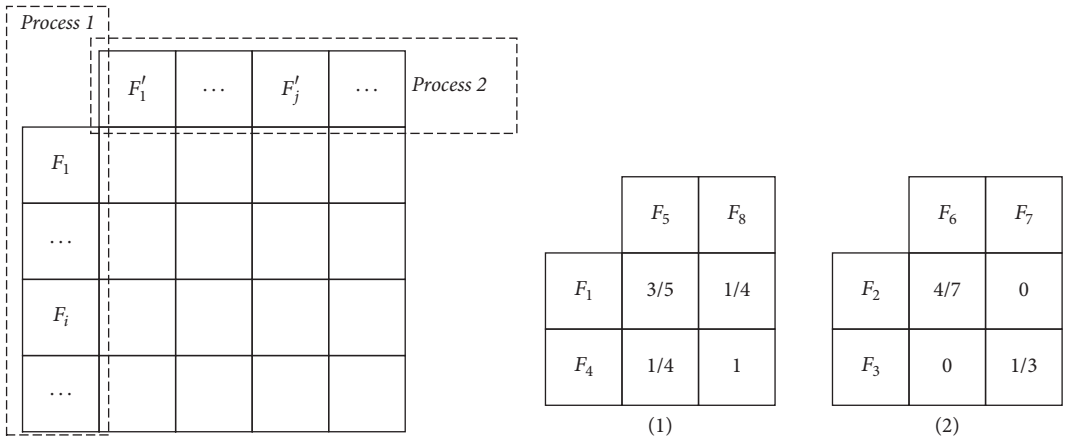
Main Idea. The main idea is that we decompose the two TPSTs into several fragments that are defined in Definition 2; the difference is computed based on the mapped fragments and the mapped nodes. In terms of implementation, there are three steps to compute the difference between two TPSTs: (1) fragment mapping: decomposing two TPSTs into several fragments and determine their mapped fragment pairs; (2) node mapping: finding out the mapped nodes based on the mapped pairs of fragments; (3) edit script generation: generating the edit script between two TPSTs based on the mapped fragments and nodes.

Algorithm. The overall pseudo code is shown in Algorithm 5, where the inputs are two TPSTs that correspond with two process models, and the output is an edit script which is regarded as their difference. Firstly, two TPSTs are decomposed into several fragments, respectively; the optimal fragment mapping combination that consists of a series of mapped fragment pairs is found by *Fragment_Mapping* (line (1)). Then, the mapped nodes in each mapped fragment pairs are found by *Node_Mapping* (line (2)). Finally, the edit script is generated based on the mapped nodes and fragment pairs by *EditScript_Generation* (line (3)).

Next we present the implementation of the three phases in Sections 4.1–4.3, and the complexity of difference detection algorithm is given in Section 4.4.

4.1. Phase 1: Fragment Mapping

Main Idea. Directly detecting difference between two trees is complicated; for example, computing the edit distance between two trees is NP-complete. Thus, we adopt the divide and conquer strategy to reduce the complexity of this problem, which can improve the efficiency of the difference detection. We first decompose two TPSTs into several fragments. Then, the similarity scores of all possible mapped fragment pairs from different TPSTs are calculated according to the *fragment mapping rules*, which are defined in Definitions 9 and 10. Next, a table called *Fragment_Mapping_Table* (Definition 11) is created based on these similarity scores, and the mapping fragment combination with the highest



(a) Fragment mapping table

(b) Example of fragment mapping table

Figure 7: Fragment mapping table.

sum similarity score is found, that is, the optimal mapping fragment combination.

Definition 9 (unordered fragment mapping). Let F_1 and F_2 be two unordered fragments; their similarity score is computed according to (1), where the mapped nodes are the intersection nodes of F_1 and F_2; that is, $MapNodes = node_1 \cap node_2$.

For example, there are two unordered fragments F_1 and F_2 in Figure 6(a). $|MapNodes| = |node_1 \cap node_2| = |\{XOR, A, B, C\}| = 4$, $\mathrm{Sim}(F_1, F_2) = 1$. That is to say, F_1 and F_2 are identical.

Definition 10 (ordered fragment mapping). Let F_1 and F_2 be two ordered fragments and let SN_1 and SN_2 be their corresponding node sequences, respectively. The similarity score of F_1 and F_2 is computed according to (1). The mapped nodes are the longest common node subsequence of SN_1 and SN_2; for example, $MapNodes = LCNS(SN_1, SN_2)$.

In Definition 10, we do not consider the mapping for the leaf nodes that are route nodes. It is because a route node represents its corresponding fragment, so the similarity score of two route nodes is equal to the similarity of their corresponding fragments. In order to compute the similarity score of two route nodes, we need to iterate the TPSTs to find out their corresponding fragments. Once there exists route node in the found fragment, we need to continue to iterate the TPSTs and it stops until the leaf nodes of a fragment are all task nodes. In this way, the computing time increases dramatically. To improve the efficiency, we just consider the mapping for the leaf nodes that are task nodes when mapping two fragments.

For example, in Figure 6(b), F_1 is the ordered fragment of *Process 1* and F_5 and F_8 are the ordered fragments of *Process 2* in Figure 5. The mapped nodes of F_1 and F_5 are $MapNodes = LCNS(SN_1, SN_2) = \{Sequence, A, B\}$; thus, their similarity score is $\mathrm{Sim}(F_1, F_5) = (3 \times 2)/(5 + 5) = 0.6$. For F_1 and F_8, their mapped nodes are $MapNodes = LCNS(SN_1, SN_8) = \{Sequence\}$ and their similarity score is $\mathrm{Sim}(F_1, F_8) = (1 \times 2)/(5 + 3) = 0.25$.

Definition 11 (fragment mapping table). Let $\{F_1, \ldots, F_i, \ldots, F_m\}$ $(1 \le i \le m)$ and $\{F_1', \ldots, F_j', \ldots, F_n'\}$ $(1 \le j \le n)$ be the fragments of two process models *Process 1* and *Process 2,* respectively. A fragment mapping table *Fragment_Mapping_Table* with m rows and n columns is built, and the value of $Fragment\_Mapping\_Table[i][j]$ is the similarity score of F_i and F_j', as shown in Figure 7(a).

Since the fragments of a TPST can be divided into two types, we create two types of fragment mapping table to determine the mapped fragments of two TPSTs: ordered fragment mapping table and unordered fragment mapping table: *Order_fragmentMT Unorder_fragmentMT*. Only the fragments with the same type have potential to be mapped; that is, ordered fragment can only map with ordered fragment, and the same to the unordered fragments. Taking Figure 5 as an example, we create an ordered and an unordered fragment mapping table for *Process 1* and *Process 2*, as shown in Figure 7(b). (1) is the ordered fragment mapping table, where the ordered fragments of *Process 1* and *Process 2* are F_1 and F_4, F_5, and F_8, respectively, and "3/5" is the similarity score of F_1 and F_5. Similarly, (2) is their unordered fragment mapping table.

Algorithm. In this phase, the inputs are two TPSTs t_1 and t_2 and the output is the optimal fragment mapping combination M_F. Firstly, the route nodes of two TPSTs are determined by tree traversal. For each route node, it and its adjacent child nodes form a fragment. Accordingly, the type of fragment is determined according to the type of the route node. Then we get two ordered fragment sets and two unordered fragment sets from two TPSTs: $order\_F\_t_1$ and $order\_F\_t_2$, $unorder\_F\_t_1$ and $unorder\_F\_t_2$. Next, the *Order_fragmentMT* and *Unorder_fragmentMT* are initialized, respectively, where *Order_fragmentMT* records the similarity scores of all possible pairs fragment that one is from $order\_F\_t_1$ and the other is from $order\_F\_t_2$. *Unorder_fragmentMT* is created in the same way. Finally, to find the optimal ordered or unordered fragment mapping combinations that have the maximum sum of the similarity score, we turn to the Hungarian algorithm [14, 15]. The

union of ordered and unordered fragment mapping combinations is the final result of mapped fragments between two TPSTs.

Example. Figure 7(b) is the initial results of *Order_fragmentMT* and *Unorder_fragmentMT* for two TPSTs of Figure 5. After using the Hungarian algorithm twice, we obtain the optimal ordered mapping combination, $\{(F_1, F_5), (F_4, F_8)\}$, and the optimal unordered mapping combination, $\{(F_2, F_6), (F_3, F_7)\}$. Finally, the overall optimal fragment mapping combination is $M_F = \{\{(F_1, F_5), (F_4, F_8)\}, \{(F_2, F_6), (F_3, F_7)\}\}$.

When mapping two fragments, we just need to consider the nodes in the fragments rather than all nodes of the process model. For example, when computing the similarity score of F_1 and F_5 in Figure 5, the nodes of F_1 and the nodes of F_5 are considered rather than considering all the nodes of *Process 1* and all the nodes of *Process 2*. In this way, the computing space is dramatically decreased, and the mapping time is accordingly reduced.

4.2. Phase 2: Node Mapping

Main Idea. In this paper, we define two types of node operations: *node delete* and *node insert*. Thus, after we determine the mapped nodes, we can judge to which node operation the remaining nodes belong to. The main idea is that we find the mapped nodes in every mapped pair of fragments, and the mapped nodes of two TPSTs are the union of all mapped nodes in all mapped fragment pairs. In terms of implementation, different strategies are adopted to find the mapped nodes in different types of mapped fragments. For unordered fragment, the nodes with the same labels are mapped. For ordered fragment, the nodes that meet the *LCNS* are mapped. The union of mapped nodes in unordered fragments and ordered fragments are the mapped nodes of two TPSTs.

Example. In Figure 5, (F_1, F_5) is the mapped fragment pair and their mapping detail is shown in Figure 6(b), where two nodes are mapped if there exists a line. Firstly, the pair of root node (*Sequence*, *Sequence*) is mapped since they have the same type and label. The mapped leaf nodes of F_1 and F_5 are $\{(A, A), (B, B)\}$ since they meet the *LCNS*. So the mapped nodes of F_1 and F_5 are $\{(Sequence, Sequence), (A, A), (B, B)\}$. After all mapped nodes in all mapped pairs of fragment, $\{(F_1, F_5), (F_4, F_8)\}, \{(F_2, F_6), (F_3, F_7)\}$, are found, we obtain the mapped nodes of *Process 1* and *Process 2* is $\{(P_0, P_0), (B_1, B_1), (B_2, B_2), (P_4, P_4), (A, A), (B, B), (C, C), (E, E), (F, F)\}$.

4.3. Phase 3: Edit Script Generation.

Computing difference between two process models can be roughly divided into two steps: (1) determining the similar parts, which means that these parts are unchanged between two process models; (2) describing the different parts based on the similar part, where the edit script is used. So far, we have determined the similar parts between two TPSTs, that is, the mapped fragments and nodes. Next, the difference will be computed and described.

Main Idea. The goal of this phase is to generate an edit script that can transform the original $TPST_1$ into the resulting $TPST_2$. The main idea is that we determine the operation types for the different parts in two TPSTs. For unmapped nodes, the node operation type, node delete or node insert, is determined. For mapped fragments, they need to be moved if they are in different positions. In terms of implementation, there are three steps: (1) deleting nodes: the unmapped nodes in $TPST_1$ need to be deleted; (2) inserting nodes: the unmapped nodes in $TPST_2$ need to be inserted; and (3) moving fragments: the mapped fragments with the different positions need to be moved.

The reason why we need to move a fragment is that we have not considered the position of the fragment when mapping the fragments, which may lead to the result that two fragments with a different position can be mapped. In some existing methods, they map the nodes of two trees by using the strategy called top-down [16] or maximum common subtree [9]; for example, a pair of child nodes can be mapped if and only if their corresponding parent nodes have been mapped. In this way, two identical fragments with different position cannot be mapped. So all nodes in one fragment are deleted and all nodes in the other fragment are inserted, which results in more edit operations. Thus, in our paper, we first map two corresponding fragments and then judge whether they have the same position; if their positions are different then the fragment needs to move.

Algorithm. The pseudo code of this phase is shown in Algorithm 6, where the inputs are two TPSTs: t_1 and t_2, their mapped node set M_N and their mapped fragment set M_F. The output is the edit script *editScript* that can transform t_1 into t_2. There are mainly three steps to generate the edit script for t_1 and t_2. (1) Node deletion: the nodes of t_1 are iterated level by level, the current node x is deleted once x does not belong to M_N, and the corresponding operation $Delete(x)$ is added to *editScript* (line (2)–line (7)). (2) Node insertion: t_2's nodes are iterated level by level, and the current node y is inserted at the same position in t_1 if y does not appear in M_N. Firstly, the parent node of y that y is going to be inserted in t_1: $Parent(y)$ is determined. Then if $Parent(y)$ is unordered, the insert position is default 0, and if it is ordered, the inserted position is to be determined, that is, which position y is going to insert as the child node of $Parent(y)$. The corresponding edit operation is recorded as $Insert(y, Parent(y), position)$ and added to *editScript* (line (8)–line (13)). (3) Fragment move: for each mapped fragment pair (f_1, f_2) of M_F, it is not necessary to move if the positions of f_1 and f_2 are identical; for example, the parent node pair of f_1 and f_2, $(parent\_f_1, parent\_f_2)$, belongs to M_N. Otherwise, f_1 needs to be moved to the position where f_2 is; for example, f_1's new parent node is the mapped node of f_2's parent node in t_1. The same as inserting a node, we need to consider the position when moving a fragment (line (14)–line (20)). Thus, $Move(f_1, parent(f_1), position)$ is added to *editScript*, where $parent(f_1)$ represents the root node that f_1 is going to move.

Example. As shown in Figure 5, F_4 and F_8 are two fragments in *Process 1* and *Process 2*, respectively, and they are identical

Input: TPST t_1, TPST t_2, mapped node set: M_N, mapped fragment set: M_F
Output: The edit script: *editScript*
(1) EditScript *editScript* = ∅;
(2) **for** each node of t_1 in level-order **do**
(3) let x be the current node;
(4) **if** x is not belong to M_N **then**
(5) add Del(x) to *editScript*;
(6) **end**
(7) **end**
(8) **for** each node of t_2 in level-order **do**
(9) let y be the current node;
(10) **if** y is not belong to M_N **then**
(11) add Insert(y, *Parent*(y), *position*) to *editScript*;
(12) **end**
(13) **end**
(14) **for** each fragment pair (f_1, f_2) of M_F **do**
(15) $parent\_f_1 = f_1$.root.parent;
(16) $parent\_f_2 = f_2$.root.parent;
(17) **if**($parent\_f_1$, $parent\_f_2$) is not belong to M_N **then**
(18) add Move(f_1, $parent\_f_2$) to *editScript*;
(19) **end**
(20) **end**
(21) return *editScript*;

Algorithm 6: EditScript_Generation.

but with the different positions. So F_4 needs to be moved as the child fragment of B_2, because B_2 is the mapped node of B_4 that is the parent node of F_8. That is, the edit operation is $Move(F_4, B_2, 0)$. The edit script of transforming *Process 1* into *Process 2* is $editScript(Process\ 1, Process\ 2) = \{Del(I), Del(J), Insert(D, B_1, 0), Move(F_4, B_2, 0)\}$.

4.4. Complexity Analysis. In this section, we analyze the time complexity of our algorithm. Let n_1 and n_2 be the number of two models' nodes, let f_1 and f_2 be the size of fragments (i.e., the number of nontask nodes), and let n be the average number of nodes of fragments in two TPSTs. In Phase 1, that is, fragment mapping, we first obtain two fragment sets of two TPSTs by hierarchical traversal, which achieves $O(n_1, n_2)$ complexity in execution time; then the Hungarian algorithm is used to find the optimum fragment mapping combination, which has the worst time complexity of $O((\min\{f_1, f_2\})^3)$. In Phase 2, that is, node mapping, we first iterate all pairs of mapped fragments, the time of mapping each pair of fragments is $O(n_2)$, and the total time of this phase is $O((\min\{f_1, f_2\}) \times n^2)$. In Phase 3, edit script generation, all nodes of two TPSTs and all their mapped fragments are iterated to generate the edit script, which spends $O(n_1 + n_2 + \min\{f_1, f_2\})$. In summary, Phase 2 spends the most time of the overall algorithm, and the total time complexity is $O((\min\{f_1, f_2\}) \times n^2)$.

5. Experiment

In this section, we evaluate the performance of our algorithm in terms of precision and efficiency. All experiments were evaluated on a machine with Intel(R) Xeon(R) CPU E5-2637, 3.50 GHz processor and 8 GB RAM, running JDK1.7, and Windows 7.

Table 1: The first part of dataset.

min/max/average place	7/175/35.496
min/max/average task	5/168/34.983
min/max/average edge	12/367/74.849

5.1. Dataset. The dataset that we used consists of two parts. (1) Based on the existing IBM dataset [17] we choose 10 process models as the *base* process models and modify them to their corresponding 9 variants by removing/inserting some nodes and some edges. In this way, we build a process repository with 100 process models. Table 1 shows the basic information of this process repository: minimum, maximum, and average number of place, task, and edge. (2) We choose 4-process model from the IBM dataset as the base process models, where they contain the following four control flow structures, respectively: Sequence, AND (parallel), *XOR* (exclusive), or AND + *XOR* (combining parallel with exclusive structures). For each base, we make some modifications on it to obtain its 5 variants without changing their structure. The modifications consist of deleting, inserting nodes, and moving fragments from the base, which are recorded as the standard edit script (SES). In this way, we create four repositories: Sequence, AND, XOR, and AND + XOR, where each repository contains 6 process models (1 base process model and its 5 variants) with the same structure. Table 2 shows the task node number of every process model in each repository, where the base process model has 160 task nodes

Table 2: The second part of dataset.

Control flow structure	*Base*	$Variant_1$	$Variant_2$	$Variant_3$	$Variant_4$	$Variant_5$
Sequence/AND/XOR/AND + XOR	160	140	120	100	80	60

and its 5 variants contain 140, 120, 100, 80, and 60 task nodes, respectively.

5.2. Quality Study. The second dataset is used to evaluate the precision of our algorithm, where the precision is computed by comparing the result of the algorithm with the standard edit script (*SES*). At first, we investigate the impact of varying task node size and fixing the structure on precision. Then the average precision is evaluated by fixing the structure.

5 edit scripts of $(base, variant_i)$ $(1 \leq i \leq 5)$ are computed, respectively, in each repository: *Sequence*, *AND*, *XOR*, and *AND + XOR* which are compared to the *SES*s and the ratios are plotted in Figure 8(a); we observe the impact of varying task node on precision by fixing the structure. The precision of computing difference between *Sequence* process models is 100%, while it is lower between the process models with other control flow structures. The reason is that there exists only one fragment in a *Sequence* process model, so the optimal mapped fragment pair between two *Sequence* process models can be definitely determined, and then the optimal mapped node pairs are found. However, a process model with *AND*, *XOR*, or *AND + XOR* structure has more than one fragment. For one fragment f_1 of a process model *Process 1*, there exist several fragments in the other process model *Process 2* that have the same similar score with f_1, which leads to the Hungarian algorithm randomly choosing one fragment of *Process 2* to map with f_1.

Taking Figure 8(c) as an example, the mapped node set of *Process 1* and *Process 2* is $M_N = \{(A, A), (B, B), (D, D), (E, E), (F, F), (G, G), (H, H), (I, I), (P_0, P_5), (P_1, P_8), (P_2, P_9), (P_3, P_6), (P_4, P_7), (B_1, B_4), (B_2, B_3))\}$. F_1 and F_2 of *Process 1* are the unordered fragments that are shown in the dotted boxes, which are the candidate fragments for mapping with F_3 and F_4 of *Process 1*. According to (1), $\mathrm{Sim}(F_1, F_3) = \mathrm{Sim}(F_1, F_4) = \mathrm{Sim}(F_2, F_3) = \mathrm{Sim}(F_2, F_4) = 1/3$, so there exist two optimal mapping fragment combinations: $M_{F1} = \{(F_1, F_3), (F_2, F_4)\}$ and $M_{F2} = \{(F_1, F_4), (F_2, F_3)\}$. Their corresponding mapped node pairs are $\{(B_1, B_3), (B_2, B_4)\}$ and $\{(B_1, B_4), (B_2, B_3)\}$, respectively. However, we have not made a strategy to select a better mapping fragment combination between several optimal ones; thus, which fragments are selected to map with B_1 and B_2 are unknown.

Overall, in Figure 8(a), the four tests, *Sequence*, *AND*, *XOR*, and *AND + XOR*, show the similar trends that with the decrease of the task number, the precision increases. It is because the mapped fragment set of two process models become smaller, which leads to the lower possibility that more than one optimal mapping fragment combination occurs. However, the precision for detecting difference between *Sequence* process models remains unchanged. The reason is mentioned above. For $variant_2$ in *XOR* repository and $variant_5$ in *AND* repository, the reason why their corresponding precisions decrease is that there exist many optimal mapping fragment combinations, and Hungarian algorithm outputs which one is unknown.

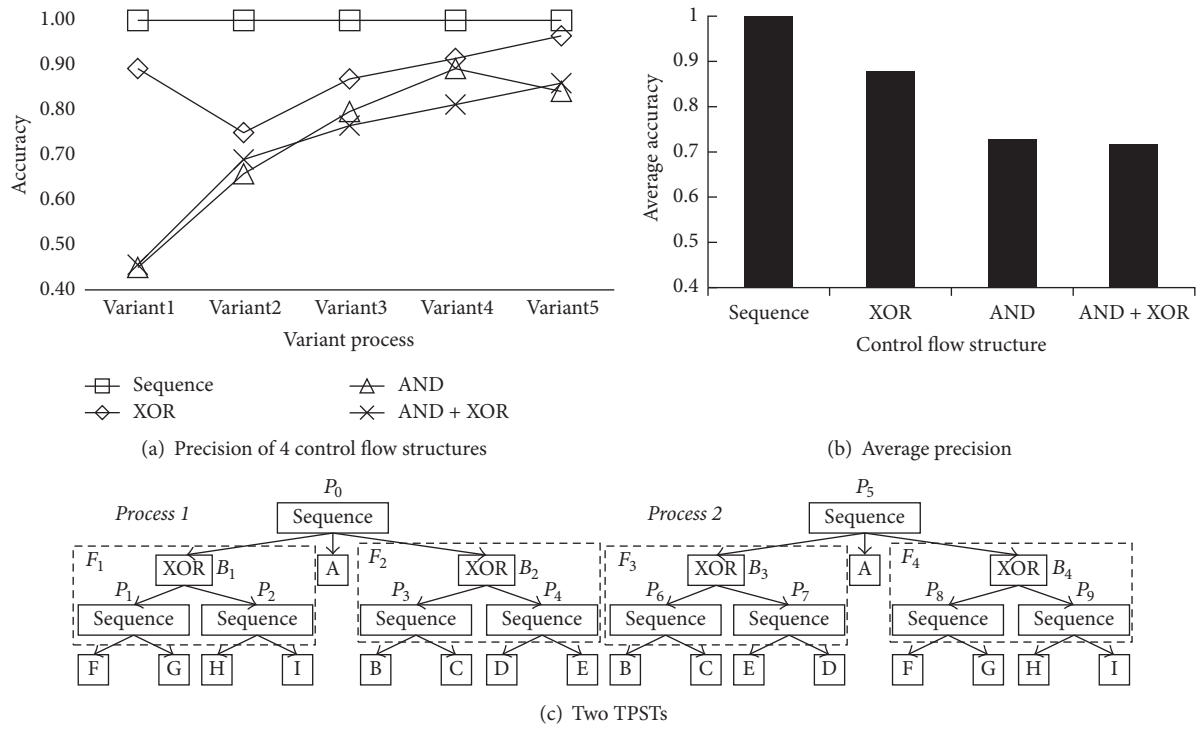
Figure 8(b) shows the average precision of four repositories with different complexity of structures and the overall average precision is higher than 70%. In summary, the precision of our algorithm is getting lower with the control flow structure getting more complicated, while it can get a better precision in the general scenario.

5.3. Efficiency Study. In this section, we conduct three kinds of experiments to evaluate the efficiency. (1) The execution time of parsing PST to TPST is evaluated, where we study the impact of changing the number of place, task, and edge, respectively, on the parsing time. (2) The execution time of detecting difference between two process models is studied, where these two process models have the similar complexity. We investigate the impact of changing one element (i.e., place, task, or edge number) of one process model on execution time by fixing the other process model. (3) The execution time of difference detection is evaluated by phases, which can be merged into two phases: node mapping that consists of mapping fragments and finding mapped nodes and edit script generation. We study the impact of varying task number on the execution time of different phases by fixing the structure.

In the first experiment, we first choose 3 sets of process models from the first part dataset. Each set contains 5 candidate process models: {(place 1, ..., place 5)}, {(task 1, ..., task 5)}, {(edge 1, ..., edge 5)}, where their place, task, and edge numbers increase progressively. Then we separately choose three target models for each set, where these three target models have the same element number (place, task, or edge number) to the first, third, and fifth models of each set. In this way, the three sets of process models are {(target 1, ..., target 3, place 1, ..., place 5)}, {(target 1, ..., target 3, task 1, ..., task 5)}, and {(target 1, ..., target 3, edge 1, ..., edge 5)}. In every set, the difference between a target model and a candidate model is computed; that is, $(\mathrm{target}_i, \mathrm{candidate}_j)$ $(1 \leqslant i \leqslant 3, 1 \leqslant j \leqslant 5)$ and the execution time of difference detection is studied.

In Figures 9(a), 9(b), and 9(c), the impacts of varying place number, task number, and edge number are studied, respectively. Overall, these three tests under different varied factors show the similar trends that our method can efficiently parse the PST to TPST in milliseconds.

With the increase of the number of place, task, or edge, the parsing time increases correspondingly. The most significant factor for impacting the parsing time is task number, and the second significant factor is place number. It is because TPST has two kinds of nodes: task node and route node. The task nodes in the process model are still the task nodes of its corresponding TPST, but the place nodes in the process model have been removed or transformed to the route nodes, so varying the number of place has smaller effect on parsing time. Varying edge numbers has the smallest impact on



(a) Precision of 4 control flow structures

(b) Average precision

(c) Two TPSTs

Figure 8: Quality study.

parsing time. It is because one edge connects two nodes; it does not change the number of nodes, but the complexity of process model also increases with the increase of edge number. In this way, it also leads to the increase of parsing time.

The dataset of the second experiment is the same as the first experiment. Figures 10(a) and 10(b) show the impact of varying place number and task number on the overall execution time, respectively, where the time increases with the increase of place or task number. There are three reasons. (1) The increase of place or task number results in the increase of fragment number, which leads to the increase of times of computing similarity score as well as its execution time. (2) The increase of place or task number results in the increase of fragment size but no new fragments, which leads to the increase of the execution time of computing similarity score between two fragments. (3) The increase of place or task number results in the increase of execution time of generating edit script.

In Figure 10(b), the execution time of computing the difference between the target model with 168 tasks and the candidate model with 99 tasks increases dramatically, while the increase of execution time is not significant for the target model with 20 tasks and the candidate model with 99 tasks. It is because the target model with 20 tasks has few fragments; even though the candidate model contains many tasks and fragments, the time of computing similarity score is small, which will not dramatically increase the execution time.

Adding an edge will cause two cases: (1) adding an edge leads to new nodes and (2) the added edge connects two existing nodes. In case (1), the execution time increases since the node number increases. In case (2), the new fragment may occur. For example, an edge is added to a process model with a single sequence structure, which may result in an extra loop structure in this process model. We can observe from Figure 10(c) that the execution time increases with the increase of edge number, which is caused by the above-mentioned two reasons. The execution time for computing the candidate model with 220 edges and three target models dramatically increases; it is because of the second reason.

In the third experiment, we use the second part of dataset to compute the difference between the *base* and its variants: $(base, variant_i)$ $(1 \leq i \leq 5)$ in every repository. Then the impact of varying task number on the execution time is investigated by fixing the structure: *Sequence*, *AND*, *XOR*, or *AND + XOR*. In Figures 11(a), 11(b), 11(c), and 11(d), we observe that the execution time of the second phase (*EditScriptGeneration*) increases dramatically and the structure is getting more complicated, while it does not increase dramatically for the first phase (*Node Mapping*). The reason is that the fragment number becomes bigger with the structure getting more complicated, which leads to the increase of computing similarity score times in *Node Mapping*. The execution time of the second phase is based on the node number and mapped fragment number of two process models. Since the task number of the four *base* models is the same, and the fragment number is so small that it has few influences on the execution time, the execution time of generating edit script does not change significantly.

In Figures 11(a)–11(d), the execution time of each phase all decreases with the decrease of task node while keeping

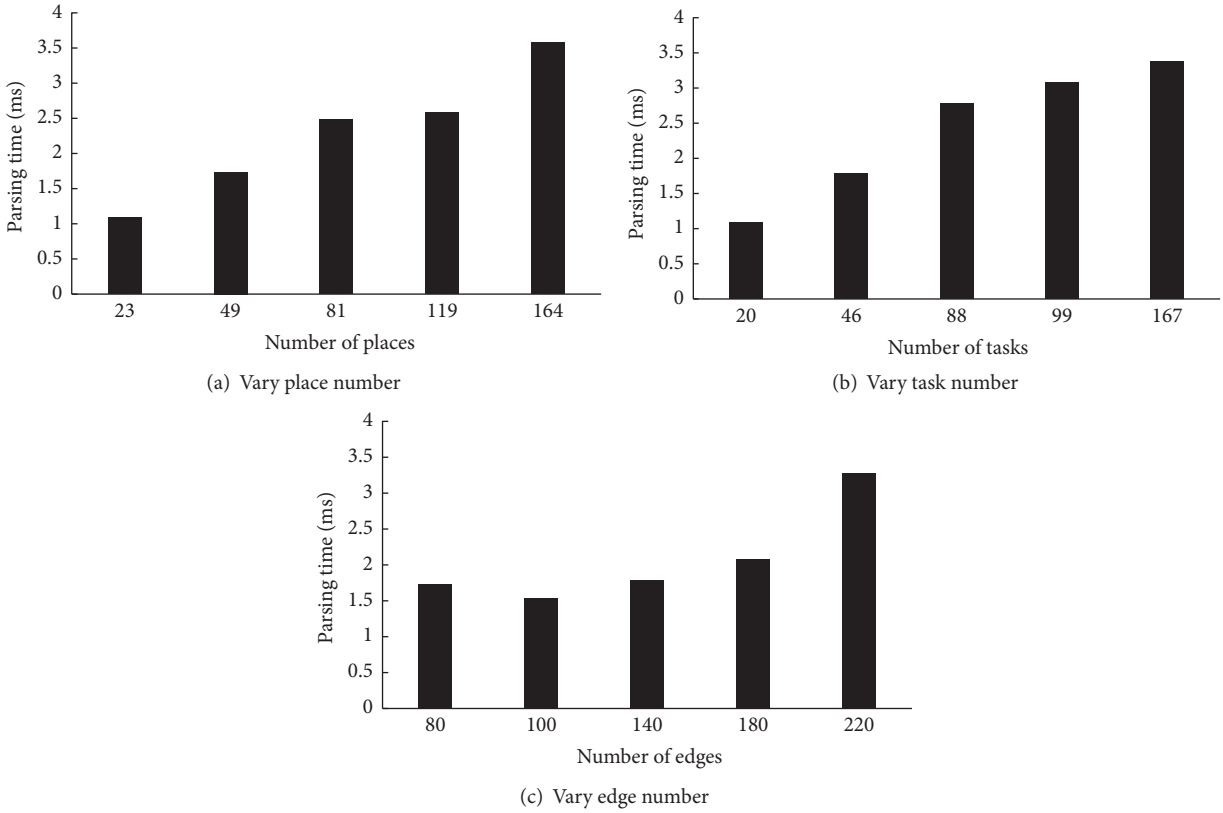


Figure 9: The first experiment of efficiency study.

the structure. The reason is analyzed in the following: taking Figure 11(a) as an example (the other three results are similar), the times of finding mapped nodes reduce with the decrease of task number. Besides, the decrease of task number results in the decrease of fragment number or fragment size, which can correspondingly lead to the decrease of computing similarity score times.

In conclusion, on the one hand, the execution time increases with the place, task, or edge number is getting larger. In particular, the following case can lead to the significant increase of execution time: changing the place, task, or edge number results in the change of structure. On the other hand, the structure is getting more complicated resulting in the increase of execution time. We can deem that our algorithm can meet the efficiency requirements of the real application scenarios according to the results of the efficiency study.

6. Related Work

The current work of difference detection can be classified into three categories. The first category is to transform the process models into their corresponding tree models, and then the difference detection is based on the tree models. Cao et al. parse the process models into their corresponding process structure tree (PST), and the difference of two process models is obtained by computing the difference between two PSTs, where they use the maximum common subtree to determine the mapped nodes [9]. But this paper does not present the implementation of parsing process models to PSTs, and finding the mapped nodes by maximum common subtree may miss other identical or similar parts of two process models.

The second category of methods performs difference detecting directly based on process models. The most related work is the method of detecting and resolving process model difference in the absence of a change log. Firstly, a process model is decomposed into several fragments with a single entry and a single exit (SESE). Secondly, the mapped nodes and the SESE fragments of two process models are determined. Finally, based on the mapped nodes and fragments, the difference of the fragments is calculated [18]. The difference between this work and our work is that we consider the similar mapping of fragments; in this way, more similar parts of two process models can be determined. Liu et al. present a method to detect the syntactic differences rather than structure differences between process models [19]. Dijkman makes a classification for the differences between process models that frequently occurred [20]. He also proposes a method to diagnose the difference between EPC models, where the exact position and type of the difference are returned [21]. Liu et al. present the definition of the structure difference of process model, and they prove that there exists this kind of differences in reality [22]. Yan et al. design an algorithm to detect the behavior difference between two process models, which achieves higher efficiency compared with the previous work [23]. Li et al. compare two process models by using high

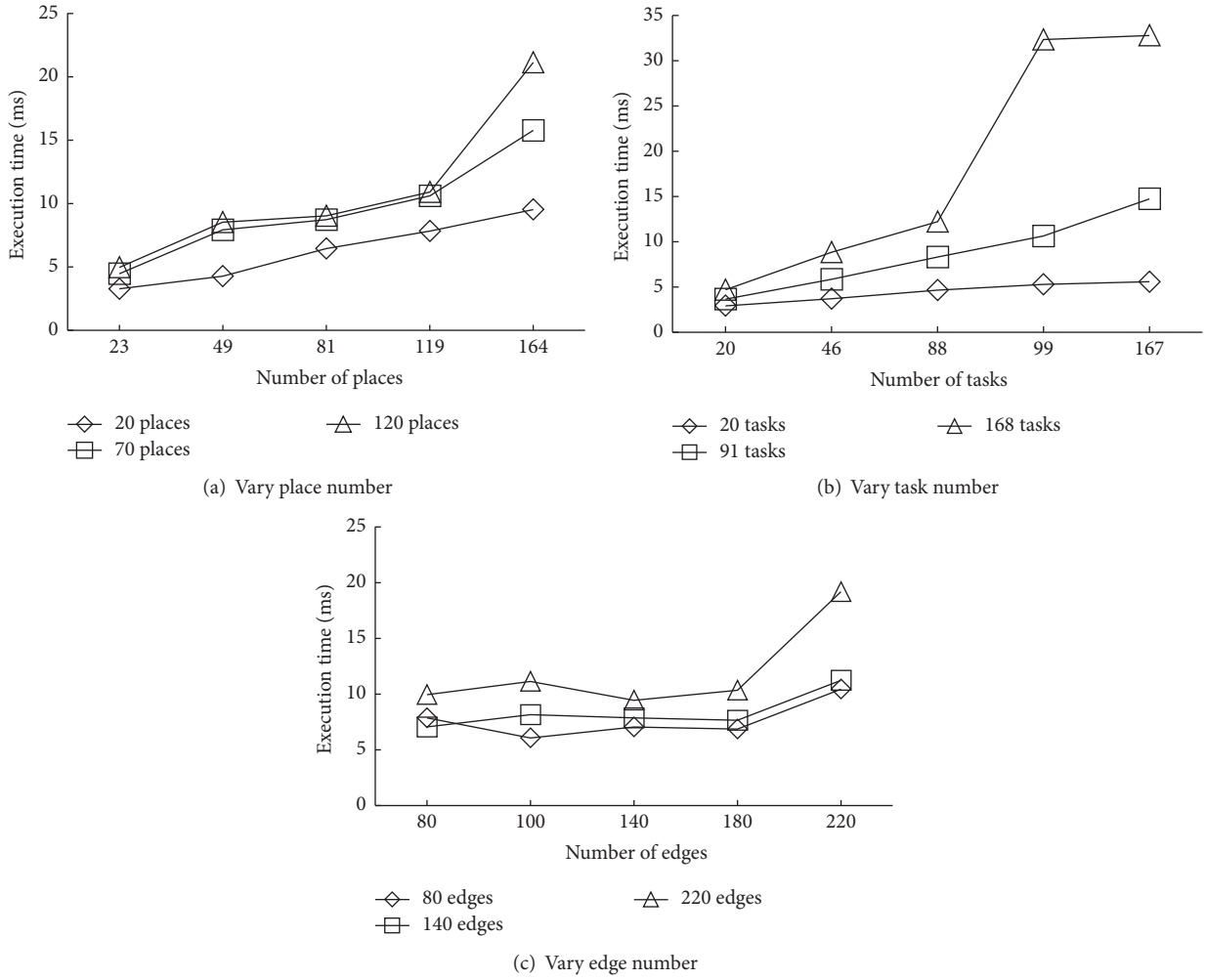


Figure 10: The second experiment of efficiency study.

level changes, like "move," in order to reduce the efforts and make the difference more understandable [24].

The last category is difference detection between structure documents, such as XML documents, where the documents are usually represented in a tree structure. Peters surveys the XML change detection algorithms, where most algorithms only consider three kinds of edit operations: node insertion, deletion, and update, and some certain properties of these algorithms are also described [25]. Al-Ekram et al. propose an algorithm with $O(n^2)$ runtime to detect changes between two versions of an XML document. They use the tree fragment mapping technique to achieve the goal of optimizing the runtime of mapping nodes and minimizing the size of edit script [26]. Cobéna et al. detect difference between XML data by trying to match more nodes. Firstly, the unchanged subtrees are determined. Based on these unchanged subtrees, more mapped nodes are found by considering ancestors and descendants of matched nodes [27]. Wang et al. use XHash and the notion of node signature to compute the difference of two XML documents that are represented to unordered trees [28]. Finis et al. propose the random walks similarity measure to find similar subtree in hierarchical data that can be represented to both ordered trees and unordered trees [29].

7. Conclusion

Nowadays, mobile workflow management system (mWfMS) is popular since the widespread use of mobile devices, which leads to large number of process models. Different locations for one business goal may result in different process models. This paper aims to detect difference between these process models. In order to solve this problem, we parse a process model to its corresponding task based process tree (TPST), and the problem of computing the difference between process models is transformed into detecting difference between TPSTs. Computing the tree edit distance between two labeled trees is NP-complete. So we use the divide and conquer strategy in our algorithm to obtain an edit script of two TPSTs that we make the cost close to minimum, where two TPSTs are decomposed into several fragments and then the corresponding mapped fragments and mapped nodes are determined. In this way, the mapping space is reduced and the mapping efficiency is improved. In experiment, we evaluate

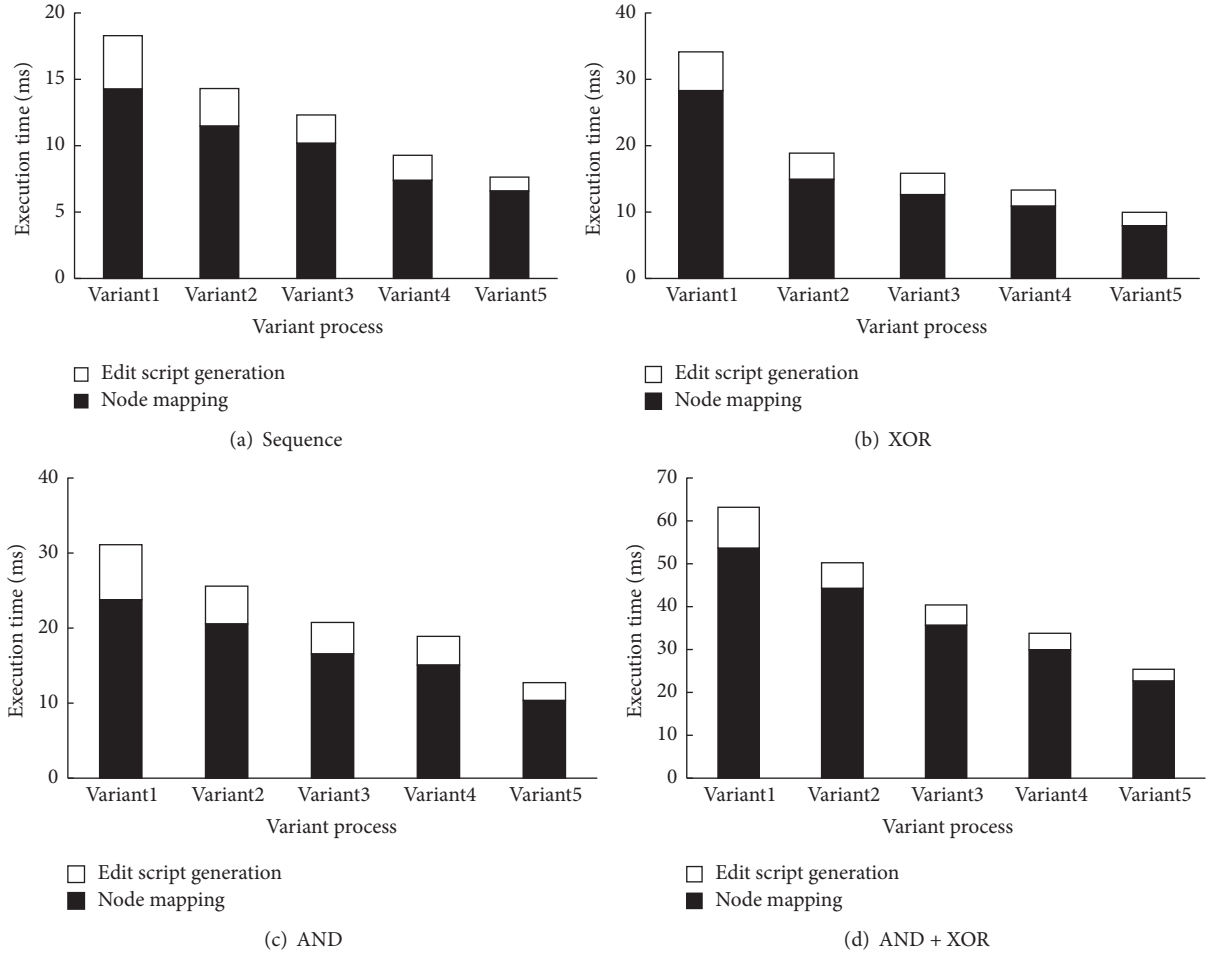


Figure 11: The third experiment of efficiency study.

the precision and execution time of our algorithm based on the real and synthetic data. The experimental results show that the precision of our algorithm is acceptable, and the execution time runs in milliseconds.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was partially supported by following foundations: National Natural Science Foundation of China (nos. 61602411 and 61572437), National Key Research & Development Program of China (no. 2016YFB1001403), Key Research and Development Project of Zhejiang Province (nos. 2015C01029, 2015C01034, and 2017C01013), and Major Science and Technology Innovation Project of Hangzhou (no. 20152011A03).

References

[1] A. Oberweis, "Person-to-application processes: workflow management," *Process-Aware Information Systems: Bridging People and Software through Process Technology*, pp. 21–36, 2005.

[2] G. Alonso, R. Günthör, M. Kamath, D. Agrawal, A. El Abbadi, and C. Mohan, "Exotica/FMDC: a workflow management system for mobile and disconnected clients," in *Databases and Mobile Computing*, pp. 27–45, Springer, New York, NY, USA, 1996.

[3] J. Jeng, K. Huff, B. Hurwitz, H. Sinha, B. Robinson, and M. Feblowitz, "WHAM: supporting mobile workforce and applications in workflow environments," in *Proceedings of the 10th International Workshop on Research Issues in Data Engineering (RIDE '00)*, pp. 31–38, IEEE, San Diego, Calif, USA, February 2000.

[4] A. Maurino and S. Modafferi, "Workflow management in mobile environments," in *Proceedings of the International Workshop on Ubiquitous Mobile Information and Collaboration Systems (UMICS '04)*, pp. 83–95, Springer, Riga, Latvia, June 2004.

[5] M. Decker, P. Stürzel, S. Klink, and A. Oberweis, "Location constraints for mobile workflows," in *Proceedings of the International Conference on Techniques and Applications for Mobile Commerce (TAMoCo '09)*, pp. 93–102, Mérida, Spain, September 2009.

[6] M. Decker, "A location-aware access control model for mobile workflow systems," *International Journal of Information Technology and Web Engineering (IJITWE)*, vol. 4, no. 1, pp. 50–66, 2009.

[7] H. Bunke, "On a relation between graph edit distance and maximum common subgraph," *Pattern Recognition Letters*, vol. 18, no. 8, pp. 689–694, 1997.

[8] J. Vanhatalo, H. Völzer, and J. Koehler, "The refined process structure tree," *Data and Knowledge Engineering*, vol. 68, no. 9, pp. 793–818, 2009.

[9] J. Cao, Y. Yao, and Y. Wang, "Mining change operations for workflow platform as a service," *World Wide Web*, vol. 18, no. 4, pp. 1071–1092, 2015.

[10] K. Zhang, R. Statman, and D. Shasha, "On the editing distance between unordered labeled trees," *Information Processing Letters*, vol. 42, no. 3, pp. 133–139, 1992.

[11] J. L. Peterson, *Petri net theory and the modeling of systems*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1981.

[12] M. Paterson and V. Dančík, "Longest common subsequences," in *Mathematical Foundations of Computer Science 1994: 19th International Symposium, MFCS'94 Košice, Slovakia, August 22–26, 1994 Proceedings*, vol. 841 of *Lecture Notes in Computer Science*, pp. 127–142, Springer, Berlin, Germany, 1994.

[13] L. Bergroth, H. Hakonen, and T. Raita, "A survey of longest common subsequence algorithms," in *Proceedings of the IEEE String Processing and Information Retrieval*, pp. 39–48, La Coruña, Spain, September 2000.

[14] B. Cao, J. X. Wang, J. Fan, T. Y. Dong, and J. W. Yin, "Mapping elements with the Hungarian algorithm: an efficient method for querying business process models," in *Proceedings of the IEEE International Conference on Web Services (ICWS '15)*, pp. 129–136, IEEE, July 2015.

[15] B. Cao, J. X. Wang, J. Fan, J. W. Yin, and T. Y. Dong, "Querying similar process models based on the Hungarian algorithm," *IEEE Transactions on Services Computing*, vol. 10, no. 1, pp. 121–135, 2017.

[16] Y. Wang, D. J. DeWitt, and J.-Y. Cai, "X-Diff: an effective change detection algorithm for XML documents," in *Proceedings of the 19th International Conference on Data Ingineering*, pp. 519–530, March 2003.

[17] D. Fahland, C. Favre, B. Jobstmann et al., "Instantaneous soundness checking of industrial business process models," in *Proceedings of the 7th International Conference on Business Process Management (BPM '09)*, pp. 278–293, Springer, Ulm, Germany, 2009.

[18] J. M. Küster, C. Gerth, A. Förster, and G. Engels, "Detecting and resolving process model differences in the absence of a change log," in *Proceedings of the 6th International Conference on Business Process Management (BPM '08)*, pp. 244–260, Springer, Milan, Italy, 2008.

[19] K. Liu, Z. Yan, Y. Wang, L. Wen, and J. Wang, "Efficient syntactic process difference detection using flexible feature matching," in *Asia Pacific Business Process Management*, vol. 181, pp. 103–116, Springer, Berlin, Germany, 2014.

[20] R. Dijkman, "A classification of differences between similar business processes," in *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC '07)*, p. 37, IEEE, Annapolis, Md, USA, October 2007.

[21] R. Dijkman, "Diagnosing differences between business process models," in *Proceedings of the International Conference on Business Process Management (BPM '08)*, pp. 261–277, Springer, Milan, Italy, 2008.

[22] K. Liu, Z. Yan, Y. Wang, L. Wen, and J. Wang, "Efficient syntactic process difference detection using flexible feature matching," in *Asia Pacific Business Process Management: Second Asia Pacific Conference, AP-BPM 2014, Brisbane, QLD, Australia, July 3-4, 2014. Proceedings*, vol. 181 of *Lecture Notes in Business Information Processing*, pp. 103–116, Springer International Publishing, 2014.

[23] Z. Yan, Y. Wang, L. Wen, and J. Wang, "Efficient behavioral-difference detection between business process models," in *On the Move to Meaningful Internet Systems: OTM 2014 Conferences: Confederated International Conferences: CoopIS, and ODBASE 2014, Amantea, Italy, October 27–31, 2014, Proceedings*, vol. 8841 of *Lecture Notes in Computer Science*, pp. 220–236, Springer, Berlin, Germany, 2014.

[24] C. Li, M. Reichert, and A. Wombacher, "On measuring process model similarity based on high-level change operations," in *Proceedings of the International Conference on Conceptual Modeling*, pp. 248–264, Springer, Berlin, Germany, 2008.

[25] L. Peters, "Change detection in xml trees: a survey," in *Proceedings of the 3rd Twente Student Conference on IT*, Enschede, The Netherlands, 2005.

[26] R. Al-Ekram, A. Adma, and O. Baysal, "diffx: an algorithm to detect changes in multi-version XML documents," in *Proceedings of the Conference of the Centre for Advanced Studies on Collaborative Research (CASCON '05)*, pp. 1–11, IBM Press, Toranto, Canada, October 2005.

[27] G. Cobéna, S. Abiteboul, and A. Marian, "Detecting changes in XML documents," in *Proceedings of the 18th International Conference on Data Engineering*, pp. 41–52, March 2002.

[28] Y. Wang, D. J. DeWitt, and J.-Y. Cai, "X-Diff: an effective change detection algorithm for XML documents," in *Proceedings of the Nineteenth International Conference on Data Ingineering*, pp. 519–530, Bangalore, India, March 2003.

[29] J. P. Finis, M. Raiber, N. Augsten, R. Brunel, A. Kemper, and F. Färber, "RWS-Diff: flexible and efficient change detection in hierarchical data," in *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM '13)*, pp. 339–348, ACM, San Francisco, Calif, USA, November 2013.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 1856329, 9 pages
https://doi.org/10.1155/2017/1856329

Research Article

Offloading Method for Efficient Use of Local Computational Resources in Mobile Location-Based Services Using Clouds

Yunsik Son[1] and Yangsun Lee[2]

[1]*Department of Computer Science and Engineering, Dongguk University, 3-26 Pil-dong, Jung-gu, Seoul 100-715, Republic of Korea*
[2]*Department of Computer Engineering, Seokyeong University, 16-1 Jungneung-dong, Sungbuk-ku, Seoul 136-704, Republic of Korea*

Correspondence should be addressed to Yangsun Lee; yslee@skuniv.ac.kr

Received 9 December 2016; Revised 6 February 2017; Accepted 20 February 2017; Published 14 March 2017

Academic Editor: Subramaniam Ganesan

With the development of mobile computing, location-based services (LBSs) have been developed to provide services based on location information through communication networks or the global positioning system. In recent years, LBSs have evolved into smart LBSs, which provide many services using only location information. These include basic services such as traffic, logistic, and entertainment services. However, a smart LBS may require relatively complicated operations, which may not be effectively performed by the mobile computing system. To overcome this problem, a computation offloading technique can be used to perform certain tasks on mobile devices in cloud and fog environments. Furthermore, mobile platforms exist that provide smart LBSs. The smart cross-platform is a solution based on a virtual machine (VM) that enables compatibility of content in various mobile and smart device environments. However, owing to the nature of the VM-based execution method, the execution performance is degraded compared to that of the native execution method. In this paper, we introduce a computation offloading technique that utilizes fog computing to improve the performance of VMs running on mobile devices. We applied the proposed method to smart devices with a smart VM (SVM) and HTML5 SVM to compare their performances.

1. Introduction

Advancements in mobile technology and location-based services (LBSs) have helped improve the quality of life of users and have fostered many business opportunities [1]. With the increasing use of LBSs, their value has likewise increased, and this value extends to the services to which the LBSs promote access.

In its early stages, the LBS used only simple location information. It has recently developed into an intelligent system that employs multiple types of information, such as the user's location, time, personal information, and behaviors [2]. It is thus difficult to effectively and efficiently perform services on a mobile device without considerable computing power. To solve this problem, a computation offloading technique can perform certain tasks in an alternative environment, such as a cloud or fog, instead of executing in the mobile device.

In addition, mobile services have the disadvantage of running various platform-dependent applications developed in different languages, such as C C++, Java, and Objective C. The smart cross-platform is a program that enables applications developed with C/C++, Java, and Objective C to run on various mobile devices, smart devices, and browsers that support HTML5 [3]. Nevertheless, owing to the characteristics of the virtual machine (VM), the performance of the hardware platform or browser on which the VM operates is greatly impacted by this approach, even if optimization is performed at the interpreter and code level. When running VM applications on a low-performance hardware platform—depending on the content complexity—it is difficult to ensure the quality of service (QoS) in terms of execution.

In this study, we strived to solve the above problems by using fog computing and a smart VM (SVM) platform to effectively and efficiently provide LBSs on mobile and smart devices. Unlike use of a centralized cloud, in the proposed approach, a local unit is employed to enable smart LBSs to effectively operate on a variety of platforms.

The remainder of this paper is organized as follows. In Section 2, we examine the features of an existing offloading scheme, the smart cross-platform, and the SVM. We analyze

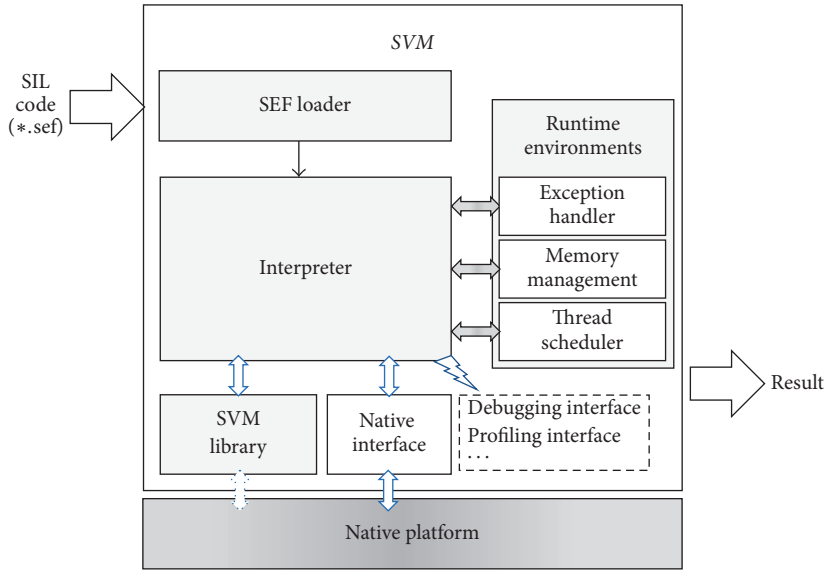


Figure 1: System configuration of the smart virtual machine.

the limitations of these respective techniques. In Section 3, we describe the proposed offloading scheme for SVMs. In Section 4, the performance of the proposed method is verified through experiments. Finally, Section 5 concludes the paper.

2. Related Works

2.1. Smart Virtual Machine Model. The SVM is a stack-based VM solution that is loaded on smart devices. It enables dynamic application programs to be downloaded and run independently of the platform. The SVM is designed to employ the Smart Intermediate Language (SIL), which can accommodate both procedural and object-oriented languages. It thus can accommodate multiple languages, such as C/C++ and Java, as well as the Objective C language used in iOS. These languages are now widely used by developers [3, 4].

The SVM system consists of three parts: a compiler that compiles application programs to create a Smart Assembly Format (SAF) file from SIL code, an assembler that converts the SAF file into a Smart Executable Format (SEF) file, and a VM that receives the SEF file and runs the program. The SVM configuration is shown in Figure 1.

2.2. HTML5 Smart Virtual Machine. HTML5 SVM is a VM-based solution that provides an integrated environment in development and execution phases by supporting both a web-based execution environment and multiple programming languages [4]. Because separate development and execution environments exist in smart devices, separate development work must be performed based on the target device and respective platform to provide a specific type of content to multiple smart device types.

HTML5 SVM can provide various contents on heterogeneous target devices with web browsers. However, it requires adequate device performance (e.g., many frames per second) to enable the smooth production of results. Thus, the performance and QoS of the given contents depend on the target device's computing power.

Figure 2 shows the system configuration of the HTML5 SVM. It basically has two layers: SVM core and SVM adaptation layer. The SVM core performs content execution. It is comprised of four components: the SEF loader, interpreter, runtime environment, and built-in library. The second layer is the SVM adaptation layer, which has six components for interfacing with target HTML5-based web browsers.

2.3. Computational Offloading. Mobile devices provide services in an approach that differs from that of traditional personal computers (PCs) because the mobile device computing power—processing speed, memory, storage space, and battery life—is limited. In particular, the LBS requires considerable processing and battery power because it provides services based on location information collected using the global positioning system (GPS) or WiFi. Moreover, in recent years, use of LBS has increasingly required more complex computations and energy to expand its services to collect and provide more complex information.

Computational offloading is a cloud computing technique that is used to run programs and provide content when there is a computing-powered-restricted environment [5–8]. Complex tasks require higher computing power. If the target device has insufficient computing power, the QoS of the provided contents/programs decreases. In this case, the offloading technique is a possible solution. Accordingly, the target device delegates complex tasks to a cloud server instead of directly executing them [9]. Figure 3 shows the proposed offloading concept.

Methods for selecting computation offloading operations are largely classified into static and dynamic approaches.

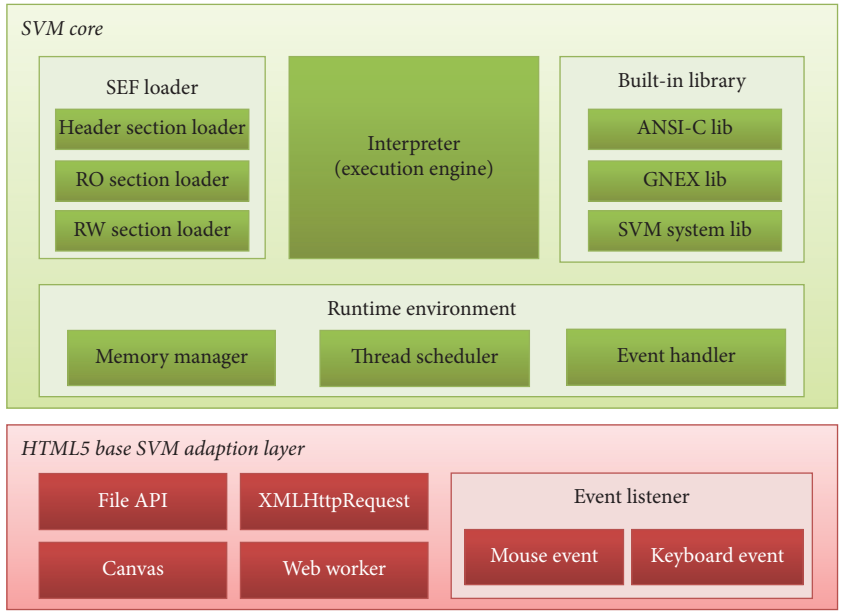


Figure 2: System configuration of the HTML5 smart virtual machine.

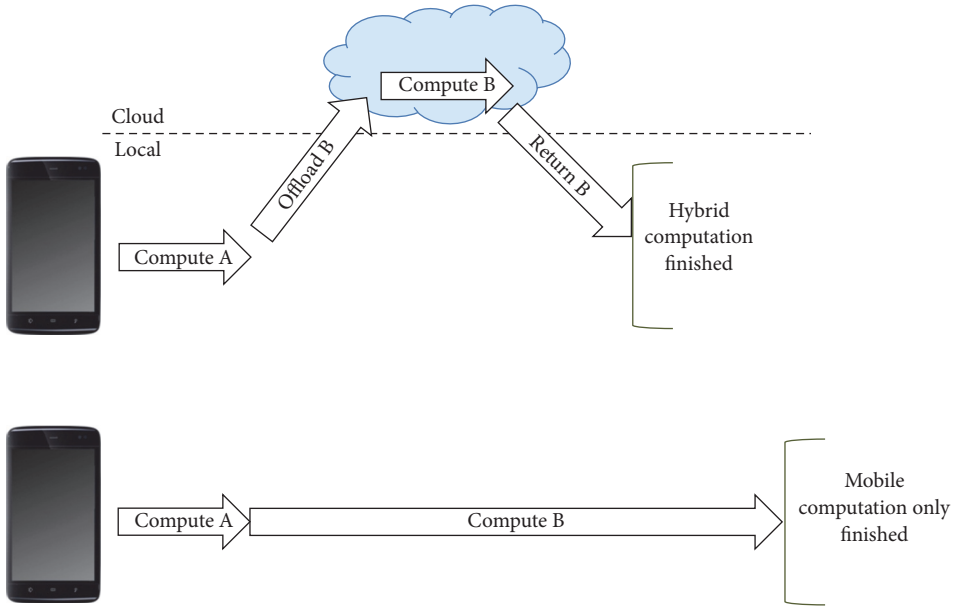


Figure 3: Computational offloading concept model.

La and Kim [10] classified offloading techniques as shown in Figure 4.

The static technique reduces the execution load by selecting the part to be offloaded during program development. The static method has the advantage of a low load in terms of cost analysis at runtime. However, the cost analysis is possible only by using predictable variables [11]. Meanwhile, the dynamic method selects the part to be offloaded with consideration of the fluctuation factors, such as the network state and remaining battery power, during execution. The dynamic method can accurately reflect the current state of the mobile device. Nevertheless, it is difficult to design a model that reflects all variables, and the required workload for the cost analysis is significant [12, 13].

Partial offloading is a method of submitting some of the work to the cloud. When a specific task is frequently used and cannot be performed in parallel, the communication costs and waiting times are increased. The full offloading method, on the other hand, addresses only the interaction with the user on the mobile device; it defers the execution to the cloud. When frequent interaction with a user occurs, synchronization problems likewise occur. Therefore, it is necessary to

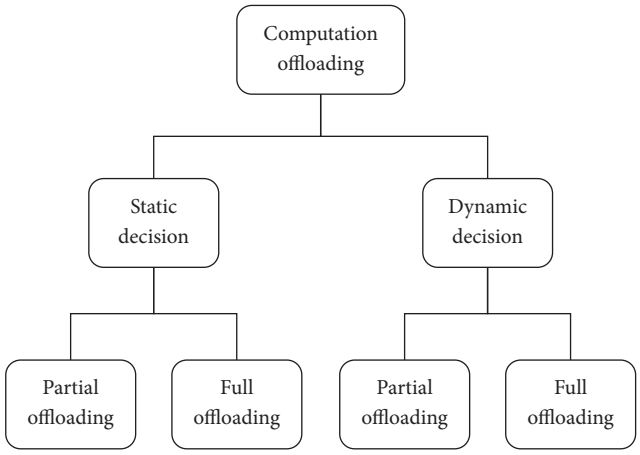


Figure 4: Classified computation offloading methods.

selectively assign an operation that is suitable for offloading to the cloud.

In the proposed approach, it was difficult to reflect real-time fluctuating factors, such as mobility and communication scenarios. Therefore, the offloading decision is based on the Mobile Augmentation Cloud Service (MACS) model [14], which estimates the code transmission cost, memory, and CPU usage for each function using the profiler at the content compile time. The offloading object determines the unit of work based on the function.

3. Offloading Module for Smart Location-Based Services

The overall system architecture for supporting smart LBs is shown in Figure 5. Each mobile device can use an appropriate fog based on its location when offloading is needed to perform the service. We designed and implemented an SVM offloading module. Offloading can improve the performance of utilizing the computing power of the cloud but may result in overhead; thus if the offloading gain is greater than the overhead cost without offloading a given task unconditionally this should be done selectively.

The offloader proposed in this paper automatically determines to offload by statically offloading through profiling of source code in the compilation step and automatically generates offloading code and transfers selected function unit work to the fog based on the location.

In the proposed approach, fog computing is employed. It is a localized service of cloud computing. It processes large amounts of data at the point of data origin, rather than at a remote server, such as centralized cloud server. Moreover, it operates on the basis of location information, similar to LBSs. Therefore, services can be provided more effectively by using fog computing when offloading complex operations in LBSs. The overhead value includes the transmission time of the data over the network, as well as the serialization/deserialization time of data transmitted locally and from the server. Analysis of the overhead is very important because the offloading performance based on the overhead can be lower than when the local operation is performed.

Figure 6 shows the offloading module structure of the SVM proposed in this paper. First, the SVM is divided into the mobile device and cloud server. Except for the thread scheduler in the runtime environment and the adaptation layer required by the actual host mobile platform, the two VMs are equivalent. In terms of content, the local SVM directly loads and executes the downloaded content. Meanwhile, the cloud SVM loads the same content as the user-executed content from the SVM application database in the cloud. This cloud refers to both the existing centralized cloud and the fog environment.

The SVM on the server has the same configuration and operation method for both cloud and fog environments. By performing the loading equivalent of VMs and content on the mobile device and the cloud, it is easy to offload the functions defined by the profiler without requiring additional work or implementing a server interface.

The process of performing the function unit offloading is as follows. First, the local SVM delegates the function unit job to the server when loading the content and calling the function (designated as offloading) during the command execution. In this process, the required context information for executing the corresponding function is extracted, serialized, and transmitted to the server. The context data refer to general data, such as the program counter, command information, and stack information used during content operation in the interpreter (the driving engine of the SVM).

The offloading module synchronizes the context between the device and server by sending the context data of the SVM to the server. The server parses the corresponding data to extract the context information required for executing the function. It performs the task of the requested function on the cloud SVM through context switching. When the function unit is finished, the changed context information is serialized and transferred to the local SVM, which reflects the changed context information to its own context to ensure that the state aligns with the result of directly executing the function.

4. Experimental Results

We applied the proposed offloading technique to SVM for a smart device and an HTML5 SVM to verify the improvement of the content execution speed. To measure the overhead caused by offloading, the local SVM was used with RaspberryPi B+, which has a Quad-Core ARM Cortex-A7 900 Mhz processor, 512 MB of memory, and the Raspbian operating system.

Figure 7 depicts the comparison of execution times before and after application of offloading through the SVM in various mobile devices and web browsers. The SVM for the smart devices was tested on an iPad2 and a Galaxy Tab 10.1. The algorithms used were the prime number, n-queen, and perfect number. HTML5 SVM was tested on a PC, iPad2, and Galaxy Tab 10.1. The prime number and n-queen algorithms were used.

The experimental results showed that the algorithm performance improved by offloading as a whole, as shown in

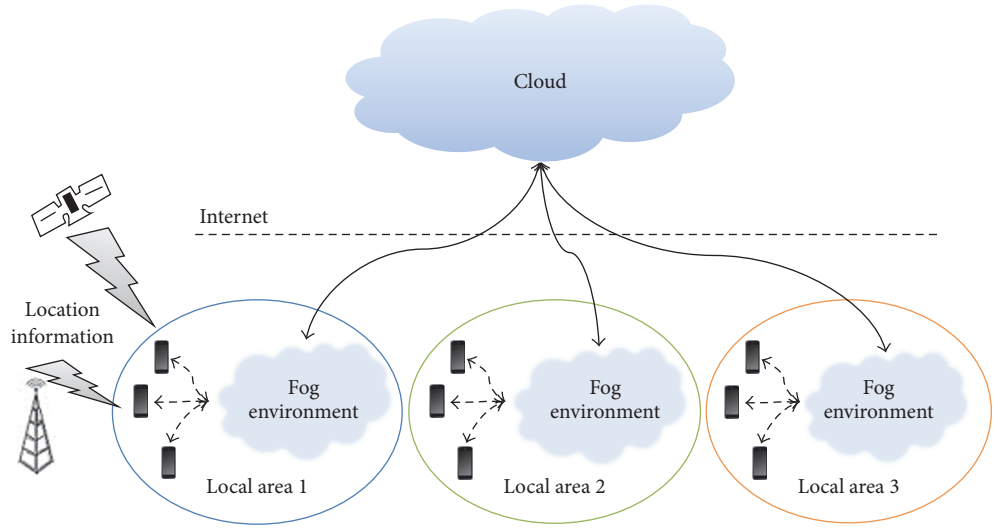


Figure 5: The system architecture to support smart LBS.

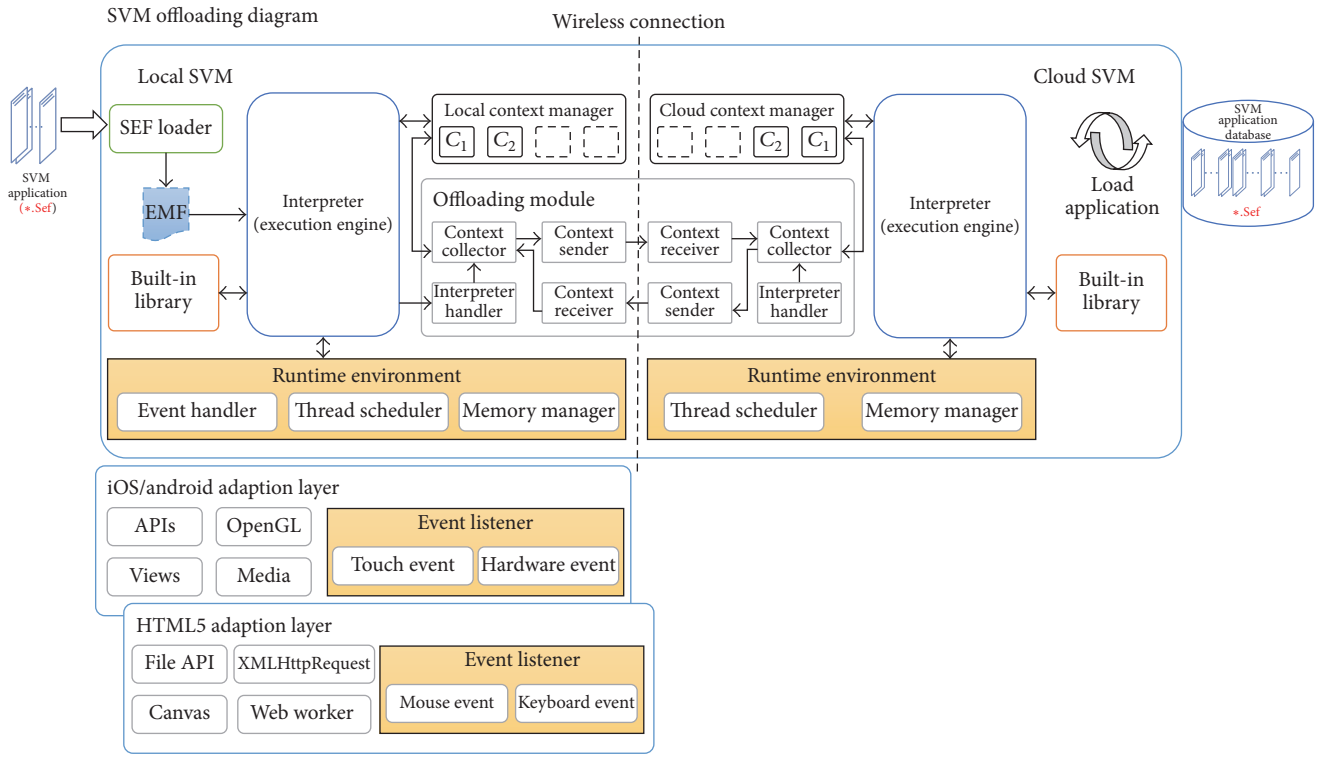


Figure 6: Proposed computational offloading modules for smart location-based service.

Figure 8. The execution time of the SVM with offloading was higher than those of the algorithm with a lower complexity and the algorithm with a higher complexity. The execution time of the SVM before offloading increased as the complexity of the algorithm increased. No significant difference was evident in terms of processing time. Therefore, the higher the algorithm complexity was, the greater the performance improvement rate was before and after offloading, as shown in Figures 7 and 8.

However, in some experiments, the offloading results showed a performance deterioration due to offloading overhead, which was incurred during the algorithm execution in the HTML5 SVM, as shown in Figure 9. The offloading overhead could be generally divided into types, such as context serialization and parsing time on the client, context serialization and parsing time on the server, transferring context data and clients, additional time incurred due to context switching, and function loading time at the server.

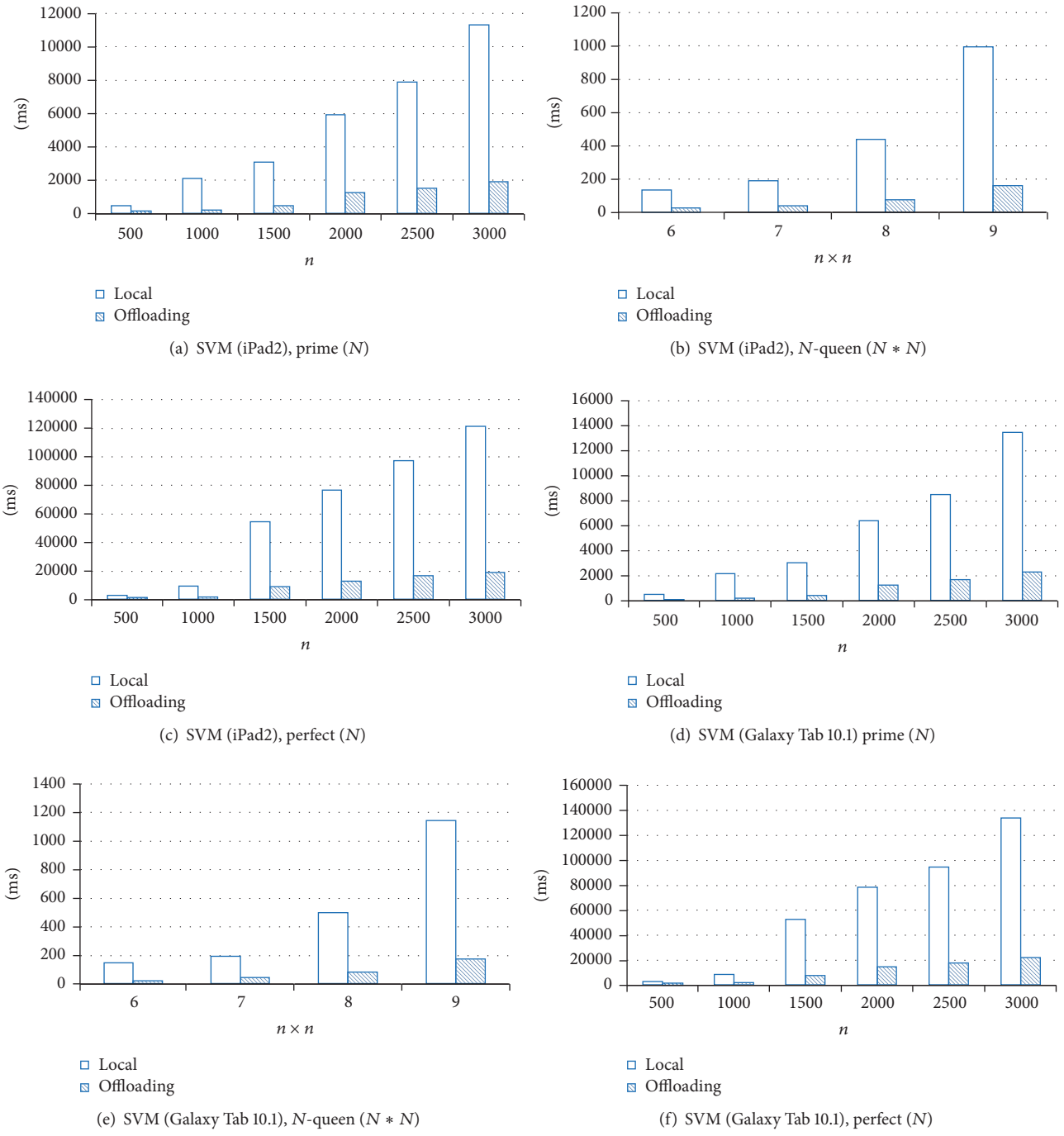
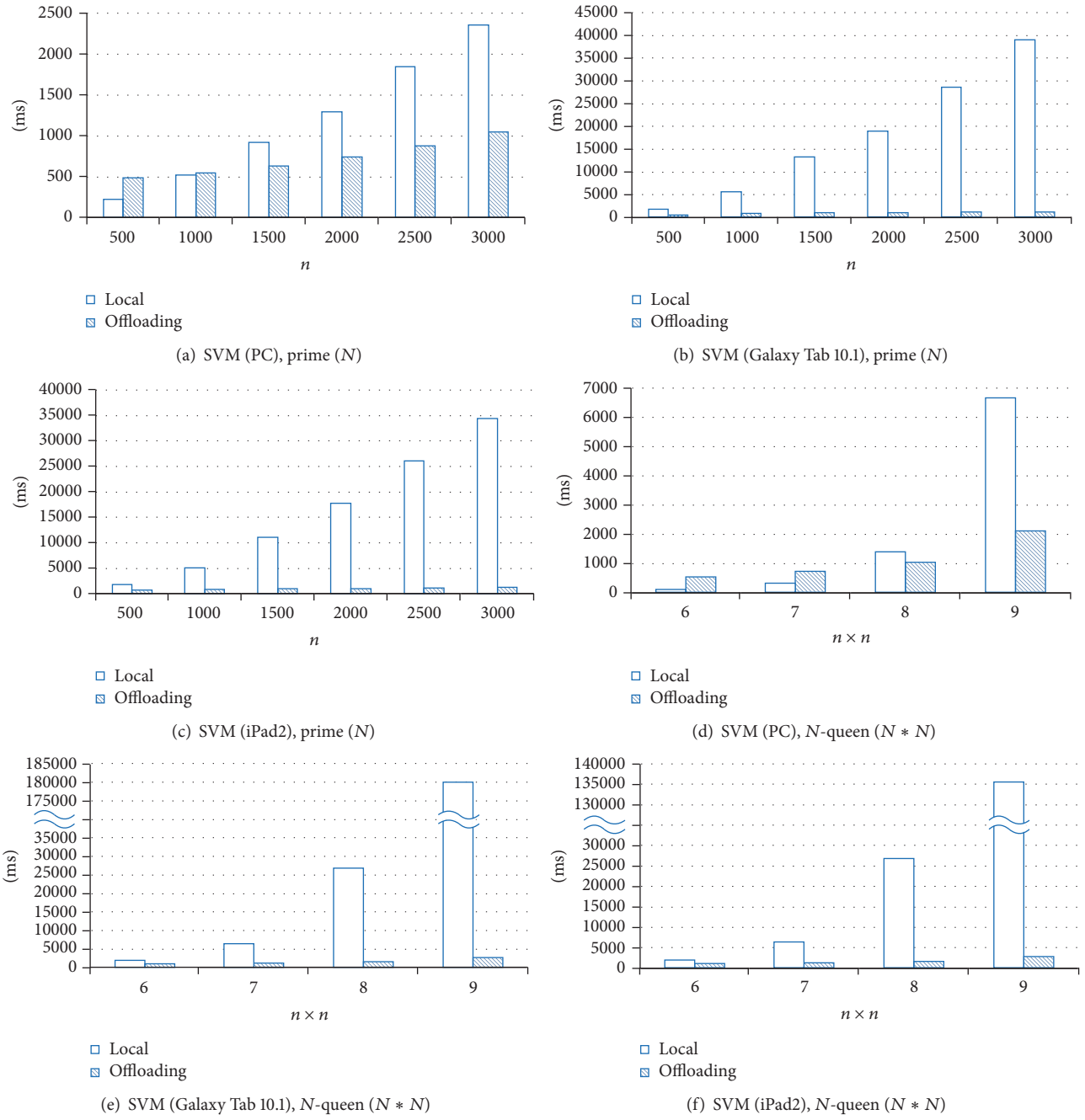


Figure 7: Comparisons of the performance evaluation using offloading in SVM on smart devices.

As a result of running the algorithm with offloading in each device, the serialization and parsing time of the server were constant. Meanwhile, the serialization and parsing time of the client differed depending on the device performance. In addition, since the additional overhead in the context switch and function loading during the operation of the client SVM depended on the device performance, it was confirmed that the time varied depending on the device.

5. Conclusions

LBSs have recently become highly available and valuable, and they now involve a variety of applications. However, with their increasing sophistication, more complicated operations are required, which causes issues on the mobile device—the main execution environment of the LBS. A complex computation load requires high computing power; thus, it is difficult



(a) SVM (PC), prime (N)

(b) SVM (Galaxy Tab 10.1), prime (N)

(c) SVM (iPad2), prime (N)

(d) SVM (PC), N-queen ($N * N$)

(e) SVM (Galaxy Tab 10.1), N-queen ($N * N$)

(f) SVM (iPad2), N-queen ($N * N$)

Figure 8: Performance evaluation of the offloading in HTML5 SVM for same algorithms with different target devices.

to perform tasks that require high computations on devices with low computing power. Offloading addresses this issue.

With offloading, the device delegates to the server a task with a high computational complexity. It sends to the server the data required for the task or the running-program context information. Because the server receives the job execution result and the changed context information, the resource consumption of the computing job can be drastically reduced based on the job characteristics. Such offloading can overcome low performance by providing high computing power of the server. Nevertheless, additional overhead is incurred on account of the communication cost of transmitting and receiving data between the device and server. The average communication cost in this study was 8 ms for the prime number algorithm (N), 7 ms for the n-queen algorithm ($N \times N$), and 10 ms for the perfect algorithm (N). Therefore, the problem could be mitigated by leveraging fog computing, which can perform intermediate processing for each region using location information.

Through the offloading technique presented in this paper, the SVM provides high computing performance from the server. It can thus perform tasks that require high computing

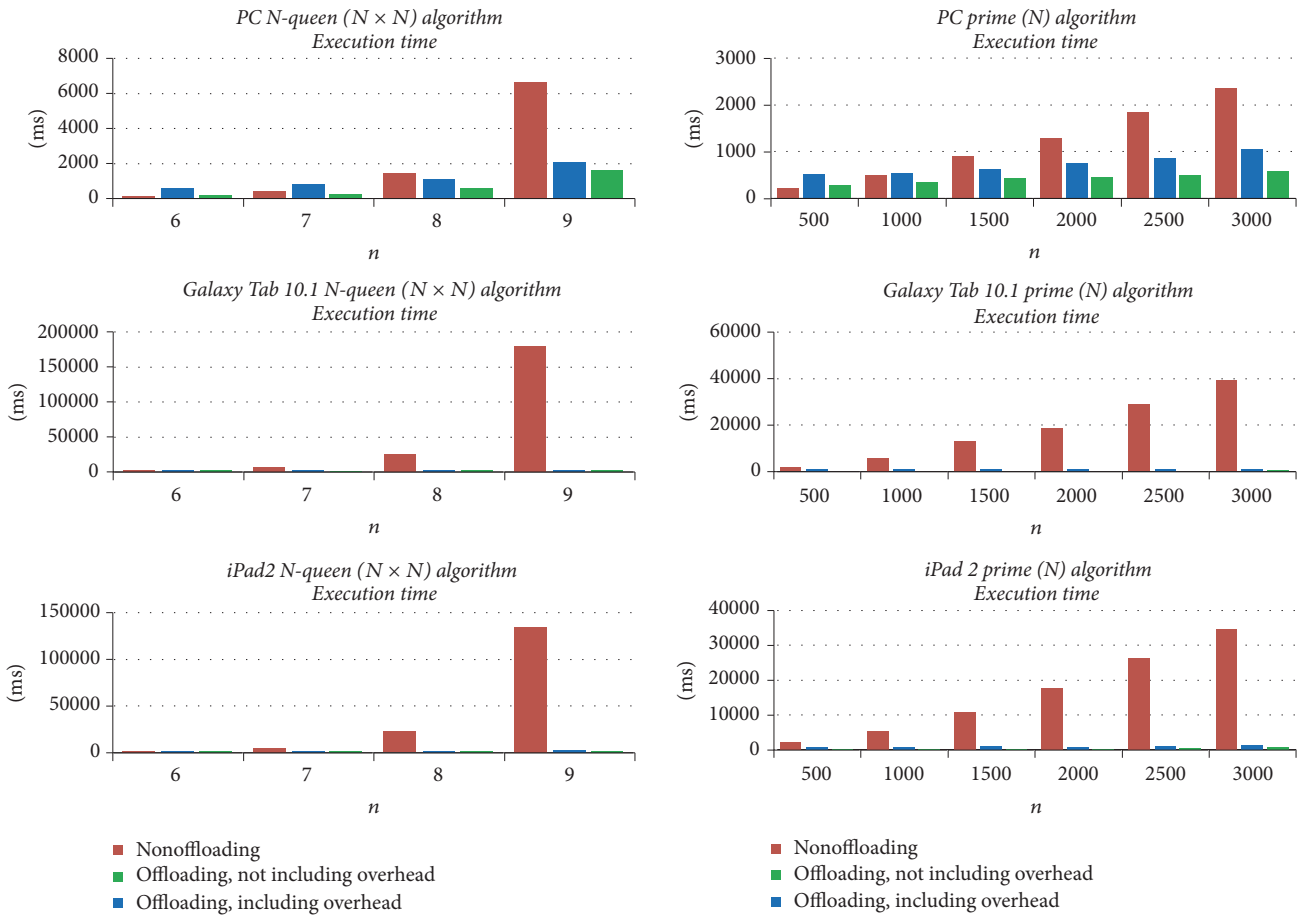


Figure 9: Offloading overhead on HTML5 SVM.

operations, even on low-performance platforms. Offloading includes context serialization and parsing time, server serialization and parsing time, and data transmitting and receiving time. Thus, because additional overhead, such as server drive time, is incurred, the offloading should be performed only when the execution time of the job to be offloaded is larger than the overhead incurred in offloading.

The SVM offloading module currently under study has a structure for delegating a task to a server through offloading during a specific function call. This call is made using precalculated profile information while the content is running. Even though indiscreet offloading is a simple task with less time than overhead, its application to offloading can be used for lower performance than the existing one. To solve this problem, the overhead caused by the offloading module of the SVM is minimized through a decision model that determines in advance the efficiency of offloading by calculating the performance difference before and after applying the offloading application. We intend to perform research to improve the performance accordingly.

Conflicts of Interest

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

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (no. 2016R1A2B4008392).

References

[1] E. Kaasinen, "User needs for location-aware mobile services," *Personal and Ubiquitous Computing*, vol. 7, no. 1, pp. 70–79, 2003.

[2] A. Pingley, Y. Wei, Z. Nan, F. Xinwen, and Z. Wei, "CAP: A context-aware privacy protection system for location-based services," in *Proceedings of the 29th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS '09)*, pp. 49–57, Montreal, Canada, June 2009.

[3] Y. S. Lee and Y. S. Son, "A study on the smart virtual machine for executing virtual machine codes on smart platforms," *International Journal of Smart Home*, vol. 6, no. 4, pp. 93–106, 2012.

[4] Y. Son, S. Oh, and Y. Lee, "Design and implementation of HTML5 based SVM for integrating runtime of smart devices and web environments," *International Journal of Smart Home*, vol. 8, no. 3, pp. 223–234, 2014.

[5] K. Yang, S. Ou, and H.-H. Chen, "On effective offloading services for resource-constrained mobile devices running heavier

mobile internet applications," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 56–63, 2008.

[6] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Networks and Applications*, vol. 18, no. 1, pp. 129–140, 2013.

[7] C. Shi, K. Habak, P. Pandurangan, M. Ammar, M. Naik, and E. Zegura, "COSMOS: computation offloading as a service for mobile devices," in *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '14)*, pp. 287–296, Philadelphia, PA, USA, August 2014.

[8] B. G. Chun, "Clonecloud: elastic execution between mobile device and cloud," in *Proceedings of the 6th ACM Conference on Computer Systems*, pp. 301–314, Salzburg, Austria, April 2011.

[9] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[10] H. J. La and S. D. Kim, "A taxonomy of offloading in mobile cloud computing," in *Proceedings of the 7th IEEE International Conference on Service-Oriented Computing and Applications (SOCA '14)*, pp. 147–153, Matsue, Japan, November 2014.

[11] C. Wang and Z. Li, "A computation offloading scheme on handheld devices," *Journal of Parallel and Distributed Computing*, vol. 64, no. 6, pp. 740–746, 2004.

[12] H.-Y. Chen, Y.-H. Lin, and C.-M. Cheng, "COCA: computation offload to clouds using AOP," in *Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid '12)*, pp. 466–473, IEEE, Ottawa, Canada, May 2012.

[13] T.-Y. Lin, T.-A. Lin, C.-H. Hsu, and C.-T. King, "Context-aware decision engine for mobile cloud offloading," in *Proceedings of the IEEE Wireless Communications and Networking Conference Workshops (WCNCW '13)*, pp. 111–116, Shanghai, China, April 2013.

[14] D. Kovachev, T. Yu, and R. Klamma, "Computation offloading from mobile devices into the cloud," in *Proceedings of the IEEE 10th International Symposium on Parallel and Distributed Processing with Applications*, pp. 784–791, 2012.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 2835163, 7 pages
https://doi.org/10.1155/2017/2835163

Research Article

Automatic Optimizer Generation Method Based on Location and Context Information to Improve Mobile Services

Yunsik Son,[1] Junho Jeong,[2] and Yangsun Lee[3]

[1]*Department of Computer Science and Engineering, Dongguk University, 3-26 Pil-dong, Jung-gu, Seoul 100-715, Republic of Korea*
[2]*Electronic Commerce Institute, Dongguk University, 123 Dongdae-ro, Gyeongju-si, Gyeongbuk 780-714, Republic of Korea*
[3]*Department of Computer Engineering, Seokyeong University, 16-1 Jungneung-Dong, Sungbuk-Ku, Seoul 136-704, Republic of Korea*

Correspondence should be addressed to Yangsun Lee; yslee@skuniv.ac.kr

Received 9 December 2016; Accepted 9 February 2017; Published 14 March 2017

Academic Editor: Byeong Ho Kang

Several location-based services (LBSs) have been recently developed for smartphones. Among these are proactive LBSs, which provide services to smartphone users by periodically collecting background logs. However, because they consume considerable battery power, they are not widely used for various LBS-based services. Battery consumption, in particular, is a significant issue on account of the characteristics of mobile systems. This problem involves a greater service restriction when performing complex operations. Therefore, to successfully enable various services based on location, this problem must be solved. In this paper, we introduce a technique to automatically generate a customized service optimizer for each application, service type, and platform using location and situation information. By using the proposed technique, energy and computing resources can be more efficiently employed for each service. Thus, users should receive more effective LBSs on mobile devices, such as smartphones.

1. Introduction

Recently, several location-based services (LBSs) have been developed for smartphones. Of these, proactive LBSs provide services to smartphone users based on periodically collected background logs. However, since these services consume considerable battery power, it is difficult to widely use them for various LBSs on account of the characteristics of mobile devices with constrained batteries. Therefore, to successfully provide various services based on localization, the problem of overcoming the limited resources of mobile devices must be solved.

In addition, mobile devices have various platforms and require different development approaches depending on the platform. The smart cross-platform is a virtual machine-based solution that can run the same content on various smart devices. If the smart cross-platform is used to provide LBSs, it is not necessary to perform separate development for each platform. However, because the smart cross-platform has a virtual machine- (VM-) based execution environment, the performance of a program running in a VM is lower than that of a general native-code base.

In this paper, we introduce a technique to automatically generate a customized optimizer for each application, service type, and platform using location and situation information. By using the proposed technique, energy and computing resources can be more efficiently employed for each service. Thus, users can receive more effective LBSs on mobile devices, such as smartphones.

The remainder of this paper is organized as follows. In Section 2, we examine context-based optimization, characteristics of the existing smart cross-platform, the smart VM, and existing optimization techniques. Section 3 describes the code optimizer generator proposed in this paper. In Section 4, we demonstrate the performance of the optimizer. Finally, in Section 5, we present our conclusions.

2. Related Works

2.1. Context-Based Optimization Algorithms. To optimize energy consumption in LBSs, Ben Abdesslem et al. [1] proposed SensLess. This algorithm uses an accelerometer to detect motion and stasis, thereby deactivating localization. Therefore, energy consumed by unnecessary localization is

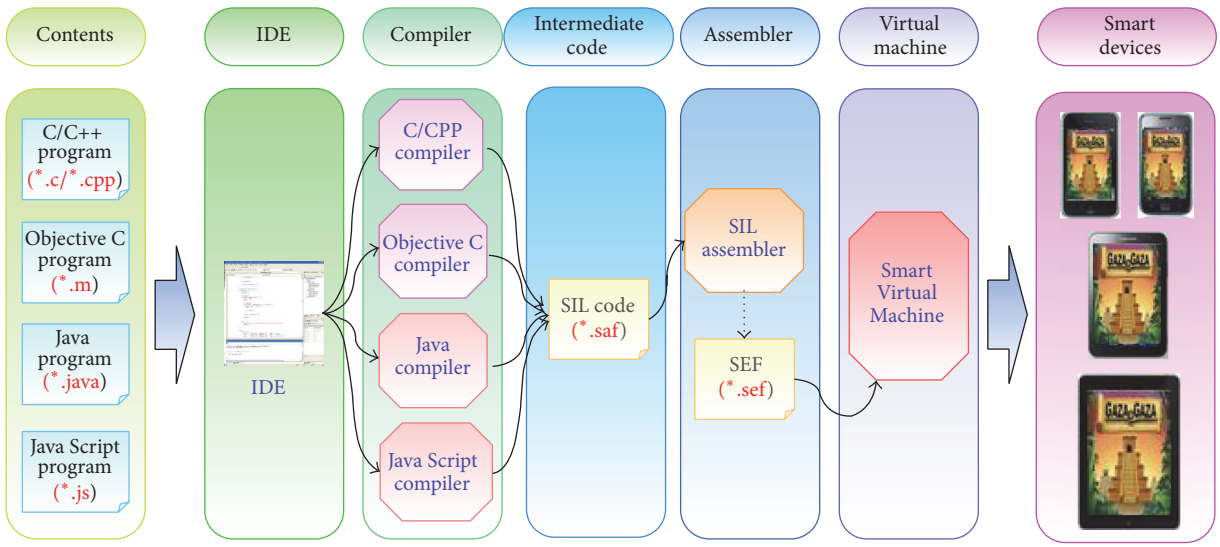


Figure 1: System model of smart cross-platform.

reduced. Nevertheless, the accelerometer must be operated at all times. Zhuang et al. [2] proposed sensor substitution and sensor suppression techniques to reduce energy consumption. The sensor substitution dynamically selects the localization method according to the automatically defined M-area whenever Global Positioning System coordinates are not available. In addition, sensor suppression technology requires continuous monitoring of the sensor, such as SensLess, by disabling localization using an accelerometer and a digital compass. Kim et al. [3] proposed the SenLoc algorithm. It prevents unnecessary energy consumption by more accurately detecting movement of a device with a three-axis accelerometer instead of simple movement. However, it can only be used in a stable Wi-Fi environment.

On the other hand, in the LBS, it is important to use a cloud to apply key personal information in accordance with each service. Therefore, effective retrieval of encrypted data is very important. Meanwhile, Fu et al. [4] proposed an effective technique for synonym-based multikeyword ranked searching over encrypted cloud data.

2.2. Virtual Machine Execution and Native Execution Method. In terms of content execution, a difference exists between the VM execution method and the native execution method. First, since the VM execution method has hardware-independent characteristics, it is easy to execute contents even if the hardware is changed. Moreover, contents can be easily transplanted to various hardware platforms. In addition, even if there is an error in the content, the target system can be continuously operated. On the other hand, owing to the software execution limitation, the performance of complex algorithms is poorer than those of native methods.

2.3. Smart Cross-Platform and SVM. Smart Virtual Machine (SVM) [5, 6] is a stack-based VM solution that can be loaded on a smart device to download and run dynamic applications on a platform-independent basis. SVM is designed to use the Smart Intermediate Language (SIL), an intermediate language that can accommodate both sequential and object-oriented languages. It has the advantage of being able to accommodate C/C++ and Java languages, as well as the Objective C language used by the iOS mobile operating system.

Figure 1 shows the overall system configuration of SVM with the smart cross-platform.

SVM and the smart cross-platform system consist of three parts: a compiler that compiles an application to generate a Standard Archive Format- (SAF-) formatted file consisting of SIL code, an assembler that converts the SAF file into an executable format, and a VM that receives and executes files in Smart Executable Format (SEF) format [7]. The smart cross-platform system is designed to be hierarchical and to minimize the burden on repurposing processes for other devices and operating environments. The SIL file from the compiler/translation process is converted into the SEF format through the assembler; SVM accepts the SEF as input and executes the program.

3. Location/Context Information-Based Automatic Optimization Model

In this paper, we introduce a technology that provides optimal execution performance by customizing various LBSs on mobile devices. The customized LBS optimization technology for mobile devices generates a pattern for each LBS using the application, context, and location information. It then optimizes the VM configuration and the application code to enable the corresponding service to be optimally executed. Figure 2 shows the LBS execution model using the proposed technique.

The service model for each mobile device using LBS is shown in Figure 3. This section introduces the service

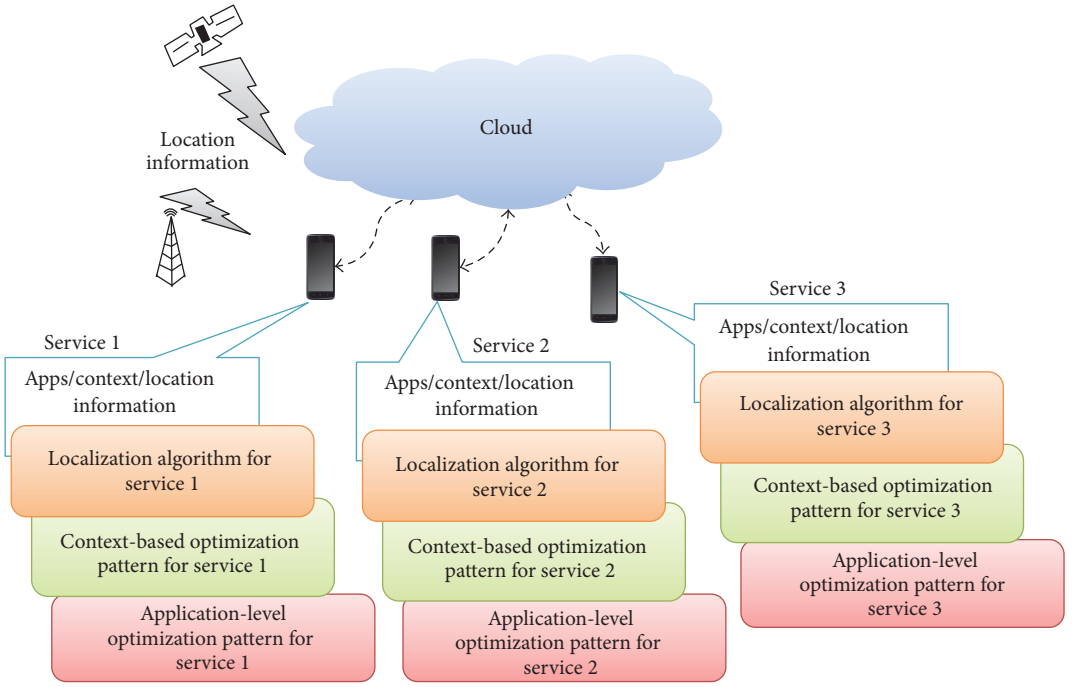


Figure 2: Customized LBSs optimization model.

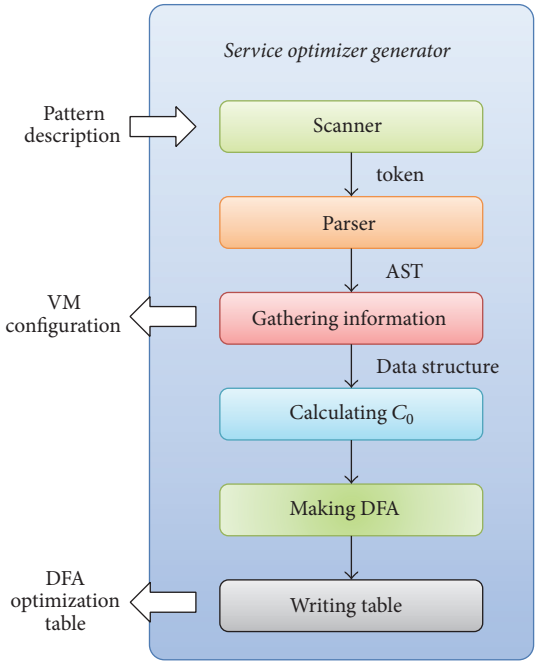


Figure 3: Configuration of the service optimizer generator.

optimizer generator (SOG) and the VM configuration for a suitable LBS algorithm on the target service with the deterministic finite automata- (DFA-) based optimization table for generating context- and application-level code optimization. The SOG proposed in this paper generates a DFA that recognizes a user-defined pattern. The optimizer optimizes the input code using it. On account of the decisive automata, the optimizer can rapidly process the input code.

3.1. Service Optimizer Generator System Model. The service optimizer generator consists of a scanner, parser, gathered information, calculated C_0, the created DFA, and a write

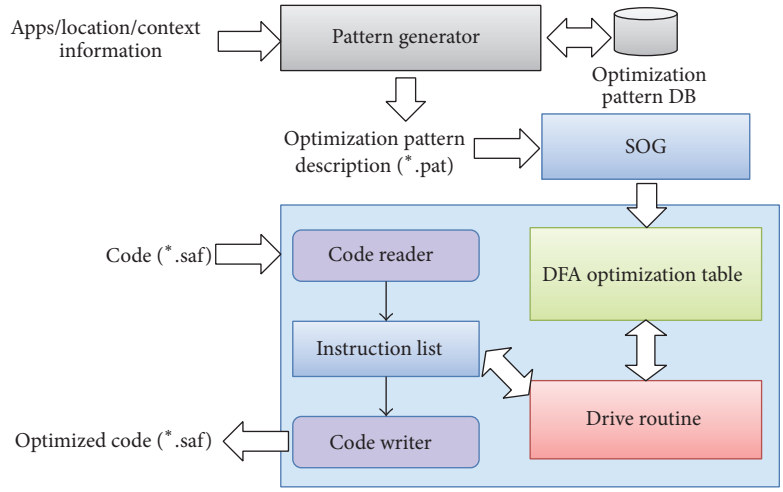


Figure 4: Code optimizer generation model using DFA optimization table.

```
optimization_pattern_description   -> '%localization_algorithm_type'
                                   '%%'
                                    context_patterns
                                   '%%'
                                    application_pattern          => OPT_PATTERN_DESCRIPTION;
context_patterns                   -> pattern_description        => CTX_PATTERN_DESCRIPTION;
application_patterns               -> pattern_description        => APP_PATTERN_DESCRIPTION;
pattern_description                -> patterns;
patterns                           -> pattern;
                                   -> pattern patterns;
                                   -> instr_code_sets
                                   ' ::=' instr_code_sets
                                   '%%' '%eval_methods' '%%'     => PATTERN;
instr_code_sets                    -> instr_code_set;
                                   -> instr_code_set instr_code_sets;
instr_code_set                     -> '%instruction' opt_parameters '/'  => INSTR_CODE_SET;
opt_parameters                     -> ;
                                   -> parameters;
parameters                         -> '%param';
                                   -> '%param' parameters;
```

Box 1: Optimization pattern description format (BNF style).

table. The scanner is the first procedure of the compiler. It creates tokens, which are syntactically significant minimum units. In other words, it receives a pattern description as an input and generates a series of tokens [8, 9]. The parser receives the token output from the scanner, checks for errors, and creates sentences according to the program syntax to generate an abstract syntax tree (AST).

The information-gathering stage involves collecting information of optimal localization methods, as well as context/application-level optimization information, to comprise the pattern description content. The content is traversed into a tree structure through the scanner and parser. The data structure is a two-dimensional layout with an integrated linked list. It is converted into a data structure wherein the pattern information and look-ahead assertion are integrated to calculate the C_0. The stage of "making" the DFA turns the calculated C_0 into a DFA format.

The DFA normalization process involves repetitions of "Calculating C_0" and "Making DFA" stages. Since the resultant DFA is normalized, optimization actions are easily performed. In other words, the DFA-type backtracking task is not necessary. Figure 4 shows the model of the code optimizer generation model using the DFA table from SOG.

3.2. Code Optimizer Pattern Description. A description of the optimization pattern is comprised of replace and pattern parts in the BNF style shown in Box 1. There are no length restrictions for the replace and pattern parts, and a single pattern comprises a single line.

```
addiv $1 $2 $3 $4   ::= 'lod.i' $5 $6 / 'lod.i' $7 $8 / 'add.i' %% $1 = $5; $2 = $6; $3 = $7; $4 = $8 %%
subiv $1 $2 $3 $4   ::= 'lod.i' $5 $6 / 'lod.i' $7 $8 / 'sub.i' %% $1 = $5; $2 = $6; $3 = $7; $4 = $8 %%
...
ldc.i.m1            ::= 'ldc.i' $1 %% $1 == −1 %%
ldc.i.0             ::= 'ldc.i' $1 %% $1 == 0 %%
...
incv.i $1 $2 $3     ::= 'lod.i' $5 $6 / 'ldc.i' $7 / 'add.i' / 'str.i' $8 $9
                        %% $5 == $8; $6 == $9; $1 = $5; $2 = $6; $3 = 7 %%
decv.i $1 $2 $3     ::= 'lod.i' $5 $6 / 'ldc.i' $7 / 'sub.i' / 'str.i' $8 $9
                        %% $5 == $8; $6 == $9; $1 = $5; $2 = $6; $3 = 7 %%
...
jgt.i $1 $2 $3 $4 $5  ::= 'lod.i' $6 $7 / 'lod.i' $8 $9 / 'le.i' / 'fjp' $10
                        %% $1 = $6; $2 = $7; $3 = $8; $4 = $9; $5 = 10 %%
jle.i $1 $2 $3      ::= 'lod.i' $6 $7 / 'lod.i' $8 $9 / 'gt.i' / 'fjp' $10
                        %% $1 = $6; $2 = $7; $3 = $8; $4 = $9; $5 = 10 %%
...
```

Box 2: Application-level pattern description example for SOG.

To distinguish the instruction code sets, the replace parts, pattern parts, and pattern descriptions are described with the separator "::=." Accordingly, the instruction code sets of the pattern part are described. The instruction code sets are described with instructions, parameters, and the separator "/." In addition, the localization algorithm type can be SensLess, SenLoc, Zhuang's localization algorithm, and so forth. These have been demonstrated in related studies [2–4].

3.3. Code Optimizer Using DFA Optimization Table. Figure 4 shows the structure of the code optimizer using the DFA optimization table, as shown in Box 2. The structure is generated from SOG through an optimization pattern description created with application, location, and context information. The code optimizer consists of five modules: a DFA optimization table, driving routine, code reader, code writer, and an instruction list.

First, when the user creates a pattern by employing the pattern specification grammar introduced in Box 1, the SOG generates a DFA-based optimization table corresponding to the pattern.

The DFA table consists of respective state and instruction sets for the code. The program input by the optimizer attempts to match with optimization instruction pattern using the instruction read by the predefined driving routine and the information recorded in the DFA optimization table. The code input unit analyzes the instruction of the inputted SAF code and converts it into a list of instructions comprised of a double connected list. In the instruction list, the optimized code table and instruction partial substitution matched by the driving routine are replaced with optimization instructions. It thus generates an optimized SAF code.

4. Experimental Results

In an experiment, the optimization code pattern for the HTML5-based smart cross-platform was entered into the implemented SOG and the generated optimizer was evaluated. Box 2 shows the optimization pattern of the SIL instruction defined according to the pattern specification grammar outlined in Box 1. The matching information of the optimization code for the parameter reduction, increase/decrease operation, and control instruction and the related conditions are provided in [8, 9].

```
...
int parsingTable[NO_STATES][NO_SYMBOLS+1] = {
        {/*** state   0 ***/
          0,   0,   0,   7,   0,   0,   6,   0,   5,   4, ...
          3,   2,   1},
        {/*** state   1 ***/
          −3,   0,   0,   0,   0,   0,   0,   0,   0,   0, ...
          0,   0,   0},
        {/*** state 2 ***/
          −1,   0,   0,   0,   0,   0,   0,   0,   0,   0, ...
          0,   0,   0},
        {/*** state 3 ***/
          −4,   0,   0,   0,   0,   0,   0,   0,   0,   0, ...
          0,   0,   0},
...
```

Box 3: Example of the generated DFA table.

Box 3 presents part of the DFA optimization table generated by SOG. As shown in the source code of the generated DFA table in Box 3, the next states to transit according to the input instruction are recorded for each target state.

Table 1 outlines data of the SIL code converted through the implemented optimizer. The input programs are a hash algorithm and tower of Hanoi algorithm source. The data show that the operator, parameter abbreviation, and control instruction for referencing the array variable were changed to the optimized SIL instruction.

The results of comparing the pattern matching performance with the string-pattern matching and tree-pattern

Table 1: Optimization results.

Hash.saf	Hash.saf (optimized)	hanoi.saf	hanoi.saf (optimized)
%Line 143:	%Line 143:	%Line 25:	%Line 25:
scanf("%s", operation);	scanf("%s", operation);	if (num == 1) move(from, to);	if (num == 1) move(from, to);
ldp	ldp	lod.i 1 12	lod.i 1 12
ldc.p @0×0247	ldc.p @0 × 0247	ldc.i 1	ldc.i.1
lda 1 0	lda 1 0	eq.i	jne.i $$0
call scanf	call scanf	fjp $$0	%Line 25:
%Line 146:	%Line 146:	%Line 25:	if (num == 1) move(from, to);
|| operation[0] == 'c')	|| operation[0] == 'c')	if (num == 1) move(from, to);	ldp
$$22: nop	$$22: nop	ldp	lod.i 1 0
lda 1 0	ldele.c 1 0 0	lod.i 1 0	lod.i 1 4
ldc.i 0	cvc.i	lod.i 1 4	call move
cvi.ui	ldc.c 97	call move	ujp $$1
cvui.p	cvc.i	ujp $$1	$$0: nop
add.p	eq.i	$$0: nop	%Line 26:
ldi.c	dup	%Line 26:	else {transfer(from, spare, to,
cvc.i	tjp $$25	else {transfer(from, spare, to,	num-1);
ldc.c 97	pop	num-1);	ldp
cvc.i	ldele.c 1 0 0	ldp	lod.i 1 0
eq.i	cvc.i	lod.i 1 0	lod.i 1 8
dup	ldc.c 98	lod.i 1 8	lod.i 1 4
tjp $$25	cvc.i	lod.i 1 4	lod.i 1 12
pop	eq.i	lod.i 1 12	ldc.i.1
lda 1 0		ldc.i 1	sub.i
ldc.i 0		sub.i	call transfer
cvi.ui		call transfer	
cvui.p			
add.p			
ldi.c			
cvc.i			
ldc.c 98			
cvc.i			
eq.i			

matching-based optimizer showed the same pattern as the DFA-based optimizer generated through SOG. Since the generated code optimizer performed pattern matching based on DFA, the input program could be rapidly analyzed by converting the SIL code optimization pattern into a deterministic finite automata.

Furthermore, the search time was decreased by 18% and 6%, respectively, compared with the conventional string-pattern matching and tree-pattern matching [10–12]. Additionally, because the DFA table was generated using the SOG, an optimizer for performing the optimization code pattern matching according to the pattern definition could be automatically generated.

5. Conclusions

In this paper, we proposed a method to generate an automatic optimizer from user-defined pattern information based on location and context information. It can effectively support LBS and improve the performance of smart cross-platform VMs. Because the pattern is represented by DFA, it shows faster optimization performance compared to existing string-pattern matching and tree-pattern matching. This makes it easy to modify or add the pattern information of the optimizer.

Moreover, the optimizer is automatically generated by the SOG; therefore, a separate optimizer correction due to the pattern change does not occur. In this study, we used DFA for fast pattern matching, which means that the expression level of the pattern had a limit that could not extend beyond the regular language. To solve this problem, we defined an area for describing a separate evaluation method in the optimization pattern.

In this paper, we introduced a technique for automatically generating a customized optimizer for each application, service type, and platform using location and context information. The proposed technique can more efficiently use more energy and computing resources for each service. Thus, users can obtain more effective LBSs on mobile devices, such as smartphones.

Our future studies will be based on the optimization pattern of the peephole level, which was limited in this study. In addition, the pattern technique that can apply the data flow will be expanded and developed. Using these studies, we will apply higher-level optimization techniques to a novel optimizer.

Disclosure

This paper was extended from the previous research paper "A Study on the Code Optimizer Generator for the Smart Cross Platform", in Advanced Science and Technology Letters, 2016 [13].

Competing Interests

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

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (no. 2013R1A2A2A01067205 and no. 2016R1A2B4008392).

References

[1] F. Ben Abdesslem, A. Phillips, and T. Henderson, "Less is more: energy-efficient mobile sensing with senseless," in *Proceedings of the 1st ACM Workshops on Networking, Systems, and Applications for Mobile Handhelds (MobiHeld '09)*, pp. 61–62, Barcelona, Spain, August 2009.

[2] Z. Zhuang, K.-H. Kim, and J. P. Singh, "Improving energy efficiency of location sensing on smartphones," in *Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '10)*, pp. 315–330, New York, NY, USA, June 2010.

[3] D. H. Kim, Y. Kim, D. Estrin, and M. B. Srivastava, "SensLoc: sensing everyday places and paths using less energy," in *Proceedings of the 8th ACM International Conference on Embedded Networked Sensor Systems (SenSys '10)*, pp. 43–56, Zurich, Switzerland, November 2010.

[4] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications*, vol. E98B, no. 1, pp. 190–200, 2015.

[5] Y. Son and Y. S. Lee, "A study on the smart virtual machine for smart devices," *Information*, vol. 16, no. 2, pp. 1465–1472, 2013.

[6] Y. Son, S. Oh, and Y. Lee, "Design and implementation of HTML5 based SVM for integrating runtime of smart devices and web environments," *International Journal of Smart Home*, vol. 8, no. 3, pp. 223–234, 2014.

[7] Y. S. Lee, J. Jeong, and Y. Son, "Design and implementation of the secure compiler and virtual machine for developing secure IoT services," *Future Generation Computer Systems*, 2016.

[8] N. Kumar and S. Hiranwal, "Current trends in the field of code optimization," in *Proceedings of the International Conference on Emerging Trends in Engineering & Management for Sustainable Development*, pp. 1–6, 2016.

[9] S. Cherubin, M. Scandale, and G. Agosta, "Stack size estimation on machine-independent intermediate code for OpenCL kernels," in *Proceedings of the the 7th Workshop on Parallel Programming and Run-Time Management Techniques for Many-core Architectures and the 5th Workshop on Design Tools and Architectures For Multicore Embedded Computing Platforms (PARMA-DITAM '16)*, pp. 1–6, Prague, Czech Republic, January 2016.

[10] A. S. Tanenbaum, H. van Staveren, and J. W. Stevenson, "Using peephole optimization on intermediate code," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 1, pp. 21–36, 1982.

[11] W. M. McKeeman, "Peephole optimization," *Communications of the ACM*, vol. 8, no. 7, pp. 443–444, 1965.

[12] C. M. Hoffmann and M. J. O'Donnell, "Pattern matching in trees," *Journal of the Association for Computing Machinery*, vol. 29, no. 1, pp. 68–95, 1982.

[13] Y. Son and Y. S. Lee, "A study on the code optimizer generator for the smart cross platform," *Advanced Science and Technology Letters*, vol. 133, pp. 138–143, 2016.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 4738340, 17 pages
https://doi.org/10.1155/2017/4738340

Research Article

Use of the Smart Store for Persuasive Marketing and Immersive Customer Experiences: A Case Study of Korean Apparel Enterprise

Hyunwoo Hwangbo,[1] Yang Sok Kim,[2] and Kyung Jin Cha[3]

[1] *Yonsei University, Seoul, Republic of Korea*
[2] *Keimyung University, Daegu, Republic of Korea*
[3] *Kangwon National University, Chuncheon, Republic of Korea*

Correspondence should be addressed to Kyung Jin Cha; kjcha7@gmail.com

Received 9 December 2016; Accepted 9 February 2017; Published 5 March 2017

Academic Editor: Subramaniam Ganesan

Information technology's introduction of online retail has deeply influenced methods of doing business. However, offline retail has not changed as radically in comparison to online retailing. Recently, studies in computer science have suggested new technology that can support offline retailers, including sensors, indoor positioning, augmented reality, vision, and interactive systems. Retailers have recently shown interest in these technologies and rapidly adopted them in order to improve operational efficiency and customer experience in their retail shops. Marketing studies also address immersive marketing that employs these technologies in order to change ways of doing offline retail business. Even though there is much discussion concerning new trends, technologies, and marketing concepts, there is, as of yet, no investigation that comprehensively explains how they can be combined together seamlessly in the real world retail environment. This paper employs the term "smart store" to indicate retail stores equipped with these new technologies and modern marketing concepts. This paper aims to summarize discussions related to smart stores and their possible applications in a real business environment. Furthermore, we present a case study of a business that applies the smart store concept to its fashion retail shops in Korea.

1. Introduction

The introduction of online retailing has deeply influenced business. In 2014, eMarketer, a marketing company, estimated that retail sales reached $22.492 trillion and will increase to $28.300 trillion in 2018. Online retail sale will account for 5.9% of the total retail market worldwide in 2014 ($1.316 trillion) and will increase significantly to 8.8% in 2018 [1]. Even though online retail has drawn attention from companies, offline retail still accounts for a large proportion of whole retail market. However, offline retail has not yet been popular area of interest for technical innovation. Recently companies and researchers have paid attention to technologies, such as sensor, indoor positioning, augmented reality, vision, and interactive interface, which helped offline retail shops to improve their service quality. In this paper, we use the term "smart store" in order to describe offline retail shops that are equipped with these technologies and that create immersive, authentic user experiences for customers. Although the idea of the "smart store" can be applied to different types of retail stores, it appears that fashion retail stores are the most appropriate place, as businesses have an interest in learning and using customers' behaviors, while customers have an interest in having a virtual experience with fashion items before physically trying them on. For this reason, many fashion companies, such as Uniqlo, Ralph Lauren, and Nike, have introduced different types of smart store applications [2–4].

Despite their importance, there are no prior studies focusing on "smart stores." This paper presents a comprehensive survey on the smart store applications within fashion retail industry and examines how various technologies can

be integrated together to provide better customer services. This consists of the following content: firstly, two different views are summarized. There are many discussions on the smart store in many academic disciplines that are largely grouped into the marketing perspective and the technology perspective. Secondly, the goals of the "smart store" are identified from the marketing perspective: customer behavior analysis and customer experience enhancement. Thirdly, the technologies that are available to support the goals of the "smart store" are examined from the marketing perspective. This paper suggests that following five techniques are essential in the smart store: sensor, indoor positioning, augmented reality, vision, and interactive interface. Fourthly, previous industry applications developed by various companies are summarized, where each application is an incidental one and not an integrated one for a business model. Fifthly, we report our experience that shows different techniques can be combined for better customer service. Finally, we suggest a framework that fits into these techniques for different business purposes.

2. Goals of Smart Store from Marketing Perspective

2.1. Customer Behavior Analysis. Most marketing scholars suggest that the goal of the "smart store" is to understand customer behaviors in brick-and-mortar stores [5]. Prior marketing studies on retail business emphasize the importance of the smart store technologies and advanced analytics as they turn in-store customer behavior data into actionable insight [6, 7]. In addition, various studies suggest that Uniqlo, Zara, and other specialty store retailers of private label apparel (SPA) brands achieve success in their business because of their manufacturing excellence, process innovation in marketing and sales, and management of consumers' dynamics with smart retail settings [8]. Uniqlo is an excellent example that demonstrates the potential of the smart store in the brick-and-mortar world. Although Uniqlo was a runner up in its e-commerce platform, it actively applied "smart store" technology, such as an in-store television and touch screen. Using marketing insights obtained from customer behavior analysis, it was able to assist customers in making appropriate purchase decisions and achieved an impressive growth in offline business. In the case of offline stores, the perspective that analyzes customer behaviors is similar to that of e-commerce sites, which track access records, clicking history, shopping cart, and series of purchases in order to understand customer behavior. In order to implement target marketing and persuasive marketing into brick-and-mortar stores, sensors, beacons, and other Internet of Everything (IoE) devices are employed to collect customer data and combine it with the purchased data.

Recently all kinds of sensors have become small and affordable; thus companies are able to use them to accumulate and collect data relevant to customer behavior in brick-and-mortar stores. Advances in mathematical modeling of face recognition and biometrics make it possible for retailers to analyze dwell time, routing, and other behavioral aspects of in-store customers and link these analytic results with purchases.

From the perspective of store operation, customer behavior analysis can be used to optimize store layout. One of the most important tasks of retail firms is to understand and apply various touch points at the moment-of-truth based on the customers' journey in the store. In relation to this, many studies from various fields have emphasized the importance of indoor location analysis [5, 9–16]. Indoor location analysis aims to provide proximity marketing because it makes it possible for retail firms to collect data of customer behaviors and in-store movements in brick-and-mortar stores, which is similar to the online web-browsing analysis in e-commerce sites. Retail firms can also use this data to conduct store visits and path analysis in order to apply analysis results to implement store layout and store management strategies. This enables retail stores to implement various layout designs that can display companies' strategic products of customers' best interests. In addition, store managers can allocate store staff more efficiently and consider time and location in which the customers are interested. Moreover, through dwell time analysis, companies can conduct marketing active ties that can increase purchase probability. Dwell time and in-between time of the actual purchases in stores are correlated [17], and such tendency strengthens primarily in regard to dwell time in specific store zones and purchases of the products within relevant domains. Therefore, store managers can increase purchasing probability not by actively responding to the customers from the beginning, but by reacting to the acquired information according to the dwell time analysis. In addition, companies can link purchase analysis with indoor location analysis to develop personalized marketing applications [18].

2.2. Customer Experiences Management (CEM). Many scholars that investigate retail industry emphasize the importance of customer experience management (CEM) [19–21]. CEM stands for internal and subjective response, which is obtained by customers via a firm's direct or indirect contact [22]. In the retail environment, macro and firm controlled factors, such as promotion, price, merchandise, supply chain, and location, can lead to enhanced customer satisfaction, more frequent shopping visits, purchases, and profits, through superior customer experience [20]. In addition, through market place rituals of six key language categories, celebrating/commemorating, gift giving, greeting, inciting, parting, and edifying store's employees, can enhance customer experience [21]. However, recent studies focus on enhancing customer satisfaction by providing new shopping trip experiences or bodily experiences to customers within stores, by using smart retail settings in addition to the methods that improve customer experience, and by improving interactions between customers and store employees [21, 23, 24]. The retail industry does not have a long history in considering technology on the full-scale as elements to enhance customer experience. Parasuraman [25] adds technology as the mediator to the two-dimensional "triangle model," which explains interactions between company, customers, and employees. This leads to the suggestion of three-dimensional "pyramid

model," after which various researchers point out that self-service technologies play influential roles to the customers' intentions [26–28] and customer satisfaction [29, 30]. In particular, Verhoef et al. [21] suggest separating the means that retailers use to provide customer experience into employee-based service and self-service technologies based service and point out the recent increase in the provision of blended services that include a mixture of the two categories. In addition, they point out that future research should proceed with investigations about how technology based service systems influence customers' shopping experience. Moreover, their research aligns with our research agenda as they claim that customers should experience new shopping experience through a smart retail setting in brick-and-mortar stores. In this regard, Schmitt [31] categorizes human experiences into five keywords: sense, feeling, thinking, acting, and relating, and proposes that they should be organized and implemented by managers in order to achieve marketing strategy and objectives. In the same line, Naylor et al. [19] discover that transformational appeals influence consumers' initial experiences by upgrading hedonic and symbolic benefits, after analyzing the influence of the consumer's actual experiences to the transformational appeals by comparing information appeals in the retail sector. Therefore, the retailers can boost their sales by adopting smart retail settings and through improved customer experience and prior purchase exchanges. In addition, the study also claims that emphasis should be put on the experience because customers are satisfied simply with a better quality of previous products. For this reason, companies' process of product consumption should emboss pleasure or sense of accomplishment as core elements.

3. Technologies for Smart Store

The transition from brick-and-mortar store to smart store is well identified in advanced retail stores, where they are based on the various technologies that support the smart store. Data collected through sensors installed inside and outside stores are used to make optimal suggestions to customers. In addition, customer behavior analysis provides customers with a virtual experience about the future of retail business. We identified the following four technologies as the fundamentals of the smart store in the fashion industry: indoor positioning, augmented reality, facial recognition, and interactive digital signage. In the following section, we examine concepts, market size, technical configurations, and practical cases.

3.1. Indoor Positioning. Indoor positioning system (IPS) refers to technology that grasps local situation of objects or people that are situated within the building through application of waves, magnetic fields, acoustic signals, or other sensory information by mobile devices [32]. Existing satellite based global positioning system (GPS) or cellular triangulation technology based location-based services (LBS) have limitations when applied in roofed buildings primarily because of the signal attenuation in accordance with construction materials [5]. However, Wi-Fi, Bluetooth, radio frequency identification (RFID), near field communication (NFC) and other similar NFC technologies, accelerometers, gyroscopes, and other recent location-sensor technologies make it possible to track user behaviors in the brick-and-mortar stores. In particular, proximity sensors, accelerometers, ambient light sensors, moisture sensors, gyroscopes, compasses, and other developments of sensor technologies play a significant role in providing detailed data of a user's behavior in a roofed building. In addition, development of microelectromechanical systems (MEMS) through sensor miniaturization advances the era of Internet of things installed in all objects. Although there is a large body of previous studies about location-based services [33, 34], recent research on indoor location applications of smart CCTV and enabling technologies has made a lot of progress [35–37]. For example, Yaeli et al. [5] analyze CCTV customer movements in the retail sector using IBM's Presence Zones and visualize optimized store layout. Meanwhile, many recent studies have studied technological elements for indoor positioning implementation. In particular, many focus on object detection and motion tracking implementation research.

There are two in-store analytics methods for tactical object detection: mobile device based radio frequency (RF) positioning techniques and video analysis. While RF positioning techniques are implemented with Wi-Fi-enabled devices or sensors and interactive mobile apps, video analysis is implemented with highly specialized cameras that monitor object movements [38]. Although there are issues of reflection and isolation because of internal walls existing within RF technology, recent interlocking of beacons installed indoors and in Bluetooth features embedded in smart phones allows RF technology to be effectively utilized to perceive user movements within a 10-meter distance. Various methods that measure people or objects location through approximating signal propagation enable vast utilization of trilateration (distance form anchors), triangulation (angle to anchors), and other inverse trigonometry based mathematical modeling. In addition, Bayesian statistical analysis [39] and other probabilistic models are recently being used.

Motion detection uses objects for dynamic territory detection on sight by comparing scene captures. The motion detection process can be divided into subprocesses, such as background subtraction, temporal differencing, and optical flow estimation [35]. Consideration and the mean shift algorithm are of vast use in object tracking. Researchers also analyze customer behavior through motion patterns classification, gained from object tracking modules, for customer path analysis [6]. Customer path analysis employs various methods, such as maximum entropy, Markov mixture models, and dynamic time warping (DTW).

Existing video sensor network systems, used for indoor positioning in public areas to assure citizens safety, also have use in tacking criminals and assure antitheft in stores. However, recent CCTV systems go beyond safety assurance. They have been utilized to investigate customers' shopping behavior [35]. In addition, recently specialized firms like American Retailnext and Euclid Analytics and Finish Walkbase have been appearing in the market. They provide retailers with visitors' analysis results, including floating

population, number of visitors, dwell time, rate of revisits, and conversion rate. Retailers can sell their products more efficiently by testing customers' convenience through eye tracking, utilization of mobile Wi-Fi, CCTV or other better measures, and analysis of in-store movement path, types of collecting customers, and floating population.

3.2. Augmented Reality. Augmented reality (AR) refers to the computer graphic technology that visualizes things that exist in the natural environment by combining computer generated sensory inputs, such as sound, video, graphics, or GPS data from the physical, real world environment. AR feels more realistic in comparison to virtual reality (VR) primarily because it looks like the overlap of the virtual with the real world through utilization of cameras, glasses, and so forth. Myron Krueger, a VR research, first introduced the concept of AR in 1970s and Thomas Caudell, a Boeing researcher, applied and used AR full-scale in 1990 [40]. In augmented reality, the computer vision techniques like video tracking are extensively used, primarily because it is important to match the real world with information provided from devices [41]. At first, fiducial markers or the optical flow of the camera image are perceived by using feature detection methods, such as corner detection, blob detection, edge detection, or thresholding [42, 43]. Then the real environment and virtual information are matched and provided to the user. Mathematical methods, such as projective geometry, geometric algebra, and nonlinear optimization have been used for this purpose. Recently a vast number of studies have been conducted to enhance the application of augmented reality based smart mirror in the retail business area [44–47].

In the beginning, the smart mirror was a mirrored digital signage, which has broadcasting, storage computing power, and network capabilities. This allows customers to take a picture of their clothes from various angles and view it in a 360-degree view, compare their appearance with other clothes, and send it to other people. Therefore, they can evaluate different products in real time. Recently, however, smart mirrors have started to incorporate augmented reality technology and add virtual fitting functions [48]. This allows stores to show off clothes in colors or sizes that are not available at the store, thereby preventing customers from leaving due to lack of interest and maximizing the customer experience.

Augmented reality was first developed for military, industrial, and medical applications, but now its use has been expanded to entertainment areas and commercial areas such as games. Recently fashion stores have begun to adopt augmented reality by utilizing augmented reality and smart mirror to provide virtual fitting and fitting recommendation services. Virtual fitting is a service that allows the user to check images visually when wearing various sizes before purchasing clothes. Customers can find out what clothes and size fit their needs by entering information about their body shape, such as their height, and their desired fit (comfort, fit, etc.). On the other hand, fitting recommendation service is a service that displays recommended clothing types and size when the customer enters information about their body shape such as height and weight. A virtual fitting service of "Memory Mirror" of Neiman Marcus is a representative example of this kind of service. At the 2014 National Retail Federation expo, MomoMi introduced "Memory Mirror" which allows for 360-degree viewing through video shooting, contains a social network service and virtual fitting function utilizing augmented reality [49]. Since then, Neiman Marcus, a luxury department store's Walnut Creek California branch, has adopted it to help customers with fitting in their stores. "Memory Mirror" is a 180 cm tall mirror that can be used to illuminate the customer's whole body. In order to use it, the customer first shoots and stores various angles such as front, side, and back views in front of the mirror for 7 seconds after fitting. The customers can compare multiple fittings with videos and photos after they have tried different clothes on. The customers can also store videos and photos in the cloud, send them to their smart phones, or share them with family, friends, and so on using social media such as Facebook. MemoMi's "Memory Mirror" also features augmented reality, showing different color clothes in real time without having to change clothes [49].

3.3. Facial Recognition. Facial recognition is a technique for identifying or verifying a person from a digital image or video frame from a video source. Facial recognition is a field of biometrics with fingerprint and eye iris recognition [50, 51]. Facial recognition is widely used in areas such as security and retail, because it has the advantage of mass identification in public places such as airports and multiplexes. The traditional facial recognition algorithms are used to identify facial features by analyzing the position, size, and shape of the face and to normalize and compress the stored face image to create face data. The most commonly used method is based on template matching techniques [52]. The recognition algorithm employs geometric approach, which recognizes faces based on the distance or shape between features, and photometric approach. Principal methods for facial recognition include principal component analysis, linear discriminant analysis, elastic bunch graph matching, hidden Markov model, and multilinear subspace learning [50, 53]. However, the recognition rate of the traditional algorithm decreases according to brightness, angle, and wearing accessories [50].

In recent years, 3-dimensional (3D) facial recognition, skin texture analysis, and thermal cameras have been introduced to support the weaknesses of traditional algorithms. 3D facial recognition technique uses 3D sensors to collect information about the shape of a face and identifies a person using features that appear on the surface of the face, such as eye sockets, nose, and chin [54, 55]. This approach has the advantage of being able to recognize faces without being affected by changes in lighting or range of viewing angles. Skin texture analysis is a technique of recognizing a person by transforming characteristic lines, patterns, and dots in human skin into a mathematical space [53]. Thermal cameras are used to remove hats, glasses, make-up, and detect actual hair shape, complementing the existing facial recognition barriers [56]. Notable software used for facial recognition include Apple's iPhoto, Google's Picasa, Adobe's Photoshop Elements, and OpenCV.

Facial recognition has been widely used for security systems in Australia, New Zealand, or the US. Recently, facial recognition techniques have been utilized in retail stores for marketing purposes through customer identification and segmentation. At the National Retail Federation (NRF) conference in 2012, Microsoft demonstrated an advertisement that offers customized products to customers in real time when customers move past the show windows using Kinect, a motion detection technology applied to the Xbox game console. It scans people passing through Kinect and suggests products that match customer's preference in Windows Embedded POSReady 7 by guessing gender, height, weight, race, age, and so on. Whole Foods has implemented the "Smarter Cart" project based on Microsoft's Kinect tablet, UPC scanner, RFID reader, and speech recognition. Whole Foods utilizes Kinect's motion capture function to allow carts to follow each other as they move through the store and apply voice commands using speech recognition technology. The reader can recognize the product name and price and create and share a shopping list. The goods' information is able to be shared automatically without paying for them. In addition to Whole Foods, Microsoft is pursuing a commerce application project based on Kinect's Windows platform for over 300 companies by tracking customer behavior in 3D and analyzing customer behavior patterns. Kinect has uses in a variety of commerce-based services, including showing advertisements that offer customized products to the customers in real time and enabling them to experience products through a virtual dressing room. San Diego based Emotient analyzes facial expressions of visitors to the store through facial recognition software and types them as joy, anger, sadness, surprise, fear, disgust, and contempt. Stores can create a salesperson response manual based on customer response and use customer feedback for marketing [57].

3.4. Interactive Digital Signage. Digital signage is digital media that provides information, entertainment, and advertisement by installing a digital display, which can be remotely controlled through a network in a public or commercial space. Media interactivity is becoming more important as consumers' media usage patterns shift from a haphazard mode to the active mode of selecting and controlling media and consuming multimedia at the same time. As the user interaction technologies, for instance, touch screen, motion detection, and image capture, evolve, the importance of interactive digital signage becomes highlighted. Interactive digital signage is an intelligent and convergent media providing contents through network. Recently, the development of digital signage has taken many forms. For example, it streams high-resolution images using large displays, provides customized advertisements through interaction with users, and supports disaster prevention through linkage with surrounding situation recognition systems. In addition, in the fashion industry, interactive digital signage is gaining attention as a customer acquisition method. It attracts customers' attention outside of the retail market, and maintains the incoming customers by drawing customers' attention.

Digital signage consists of (1) hardware for display and media player, (2) software for contents creation, distribution, management, and operation, (3) wired and wireless network for Internet and communication, (4) content authoring technology, and (5) UI & UX technology. As related technologies develop and the importance of user interaction grows, the technical elements of digital signage are rapidly changing. In terms of hardware, OLED displays, immersive displays, and large-format displays are common and the compatibility of the media player with different types of terminals is improving. Software is evolving from the limited functionality of traditional media operations to intelligent software, including hardware control, content retrieval, big data, and spatial and situational analysis. The network reflects interdevice communication and contents transmission technology. In content transmission technology, development and personalization of realistic contents based on bidirectionality are highlighted. In UI & UX, service and product development based on emotion are becoming important. As human computer interaction (HCI) technology develops, multiuser interaction via mobile devices are becoming possible.

Interactive digital signage is attempting to combine various technologies such as artificial intelligence, facial recognition, near field communication, and augmented reality. In addition, multitouch displays are used to enable multiple users at the same time. In retail stores, the use of showcases combining transparent LCD is increasing. Many companies also use NFC chips in digital signage or use QR codes or barcodes to transmit personalized information or coupon to their smartphone. In recent years, there have been cases in which the digital signage of the situation is reflected in the actual situation of the place where the digital signage is installed. Interactive Digital Billboard, an outdoor billboard launched by British Airway in November 2014, is an interactive advertisement that shows the destinations of airplanes when young children in the billboard point to airplanes when they actually fly [58]. The digital signage installed in the subway history of the train senses the entry of the train into the station and shows the hair of the model in the digital signage to be scattered in the wind of the train effectively expressing the performance of the hair care product. Interactive digital signage, in this way, combines with the latest information technology to enhance consumer engagement [59].

In the fashion industry, interactive display is implemented through interactive hangers, interactive fitting rooms, and interactive displays. Brazil's fashion retailer C&A installed a digital hanger named "FashionLike" in its stores. In clothes hangers screen, C&A's online shopping mall displays the number of customers' thumbs-up. This is similar to Facebook's Like, allowing visitors to recognize the preferences of other people's products. This provides a selection recommendation for purchasing to customers who do not have a certain preference or hesitate to purchase between several products [60]. Ralph Lauren installed an interactive fitting room called Oak Fitting Room in Oak Lab in a flagship store on Fifth Avenue in Manhattan, New York, in 2015. Customers can request different sizes by pressing the buttons installed in the fitting room, and they can recommend clothing that matches the selected item. In the interactive fitting room, when a user brings in a product they want to

Table 1: Solution-specific applied technology.

Technology	Solution				
	Smart CCTV	Smart hanger	Smart mirror	Smart show window	Smart shelf & showcase
Indoor positioning	V (motion detection, object racking)	V (beacon, accelerometer)	V (beacon)	V (beacon, motion detection)	V (beacon, accelerometer)
Augmented reality			V (camera)	V (camera)	
Facial recognition	V (facial recognition)			V (facial recognition)	
Interactive digital signage		V	V (barcode)	V (infrared touch)	V (infrared touch)

wear, they know which product they have selected through the RFID information. The mirror is also touch screen, meaning the store clerk can be asked to adjust the lighting and find a related product or if you want different size, the store clerk can be notified with the press of one button. In addition, if the customer wants to buy a product later, they are able to buy it via mobile shopping through SMS [61]. In 2012, Adidas installed an interactive digital window on its customer's smartphone without having to install a mobile application or scan a QR code in the Nürnberg NEO Label store in Germany. This storefront window is responsible for translating the virtual store. Customers can drag life-size images of products using the intuitive interface of the touch screen window to put their favorite products on their smartphones and pay for them on "Adidas NEO online." Customers can also touch up hotspots on the window to view detailed information about the product or view a moving image wearing the pertinent product [62].

4. Smart Store Framework

4.1. Solutions for Smart Stores. Company K, one of the leading companies in the Korean fashion industry, introduced three smart retail stores in the complex shopping mall in 2015. Company K has introduced six types of solutions for smart store implementation: smart CCTV, smart hanger, smart mirror, smart show windows, and smart shelf and smart showcase. Company K's smart stores typify the underlying technologies of solutions: firstly, the smart CCTV has built-in people counting line analysis and area of interest analysis based on motion detection, object tracking, and facial recognition technology. The smart hanger has a built-in accelerometer to track user's movements and to change digital display. The smart mirror provides a virtual fitting service based on augmented reality technology and provides a display in conjunction with the location of smart hanger, smart shelf, and so on. The smart show windows perform facial recognition using Kinect and beacon and provide product recommendation based on the user's gender and age in interactive display. The smart shelf and showcase track the position of the product based on the beacon with built-in accelerometer and display the user's interests on the digital signage via the infrared touch screen. Table 1 summarizes the technology utilized by the smart retail solutions in the three smart stores of Company K, and Table 2 shows its detailed functions, application solutions, and specifications.

4.2. Components of Smart Stores. Table 3 shows the smart retail settings of Company K in the flagship stores of men's clothing of S brand, women's clothing of L brand, handbags, and wallets women's accessory C brand.

L brand has installed 6 smart CCTVs for people counting, face recognition, and zone analysis. In addition, it analyzes customers' behavior through smart hangers, smart mirrors, and smart show windows and provides a new shopping experience to customers. The layout of L brand emulated applications is shown in Figure 1.

As shown in Figure 2, smart show windows are responsible for customer acquisition from outside the store, while the other five solutions perform customer retention in the store based on its installation location. We use these solutions to enhance customer behavior analysis and customer experience in terms of marketing. Smart CCTV is used for target marketing, smart show window, smart shelf, and smart showcase for experiential marketing. Meanwhile, smart hanger and smart mirror will enhance brand accessibility by providing new experiences to customers who visit first. It plays a role of increasing the purchase conversion rate. The accumulated customer behavior is used to provide personalized product recommendation and store layout optimization through data analysis.

Company K's smart stores are implemented as applications for customers and employees based on technologies such as indoor positioning, augmented reality, facial recognition, and interactive digital signage. Applications for customers include smart hangers, smart shelves and showcases, smart mirrors, smart show windows, and smartphone applications, and applications for employee include smart CCTVs, smart pads, and wearable devices. Customers who visit the store receives stimuli through shopping experiences such as viewing, listening, and touching smart retail settings. Afterwards, customers interact with applications to gain new experiences through cognition or applications that gain confidence in new information or purchases. The customer has an intention regarding shopping through the step of enjoying consciousness. This can be expressed as a customer's

Table 2: IoT technologies for K's smart stores.

IoT technology	Function	Applied solution
Movement path analysis	It captures the in-stream population of the store and collects information such as staying in a specific place for a long time or coming into a main line	√ Smart CCTV √ Wearable device
Beacon	It provides unique location signals for each product to provide relevant information from nearby equipment and your smart phone	√ Smart hanger √ Smart mirror √ Smart show window √ Smart shelf √ Smartphone application
Accelerometer	By detecting the movement of objects, the customer recognizes the action of picking up or dropping the product	√ Smart hanger √ Smart shelf √ Smart showcase
Motion detection	It provides interactive user experience with equipment through motion recognition in places where touch operation is impossible	√ Smart show window
Facial recognition	Face recognition and age/gender analysis	√ Smart CCTV √ Smart show window
Camera	It provides the customer's image to the screen and the fitting contents through the shooting/recording function	√ Smart mirror
Near infrared sensor	It provides the ability to recognize the presence or absence of customers in front of objects	√ Smart mirror
Barcode sensor	It recognizes the barcode in the tag of the product and provides the information of the product	√ Smart mirror
IR touch sensor	Through a touch function of the display glass, it provides the operation function that allows the user to view the information of the product even if the customer does not take out the product	√ Smart showcase
iWatch	Receiving a variety of notifications via watch	√ Wearable device
Smart pad	In addition to simple alarms, it provides services that enable promoting products in connection with things	√ Smart pad
Integrated control solution	Remotely managing various Internet of things equipment and distributing and operating each content	√ Smart CCTV √ Smart hanger √ Smart mirror √ Smart show window √ Smart shelf √ Wearable device √ Smart pad

Table 3: Example of K's smart stores.

Brand	Solution				
	Smart CCTV	Smart hanger	Smart mirror	Smart show window	Smart shelf & showcase
S brand (menswear)	V	V	V	V	
L brand (petticoat)	V	V	V	V	
C brand (accessories)	V		V		V

inquiry about the product to purchase, a revisit to find the store again, and a purchase to buy the goods. A series of decision-related processes can be interpreted through the Stimulus-Organism-Response (SOR) framework [63].

Customers who visit the store will be able to get detailed information about the product, such as material, price, and review, on the display installed in the vicinity as they touch, browse, and move items in smart hanger or smart shelf and showcase. Customers can use the smart mirror to compare various products at the same time through virtual fitting with 360-degree vogue and augmented reality, or to assist with purchase decisions by communicating with neighboring acquaintances. The show window's gamification function stops the customer from stepping out of the store, and the display window of the show window is turned off. In addition, smartphone applications enable location-based marketing by sending coupons or event information when a customer is near a store, allowing customers can use them. It is possible to experience Omnichannel marketing by storing items of interest in the offline store in the shopping cart of the online shopping mall or by inquiring where the items of interest in the online shopping mall are located in the offline store. Retail applications provide cognition to the demanding shopper and consciousness to the entertainment shopper to guide the customer's purchase and related decisions. The ultimate goal of a smart store in the retail industry is to attract its customers to purchase products. The purchase process in the

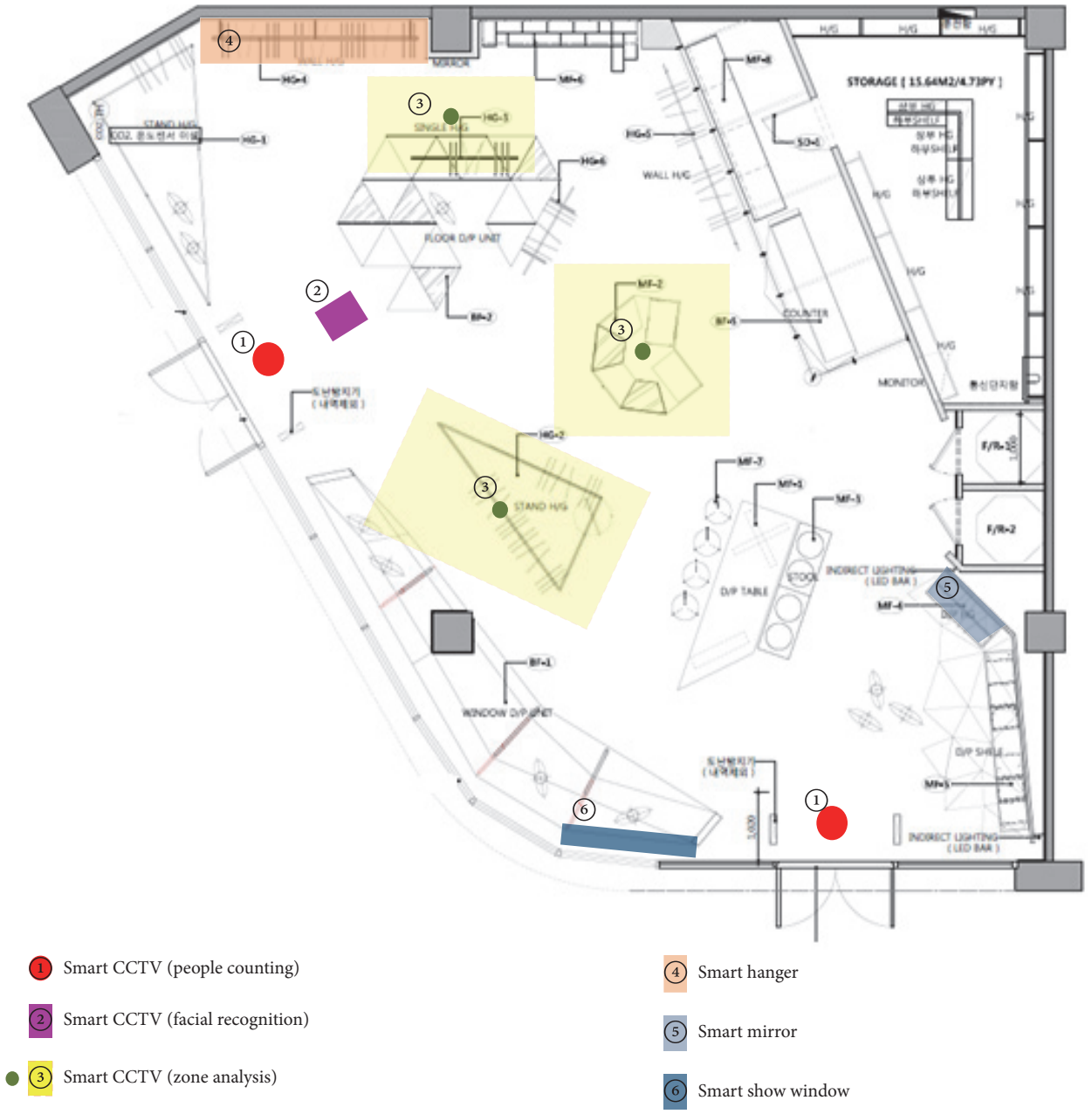


Figure 1: Layout of L brand's smart store.

marketing perspective consists of three activities: consulting the customer to have intention to purchase, inducing the customers to revisit the store, and persuading the customers to purchase the product. This can be measured by actual resource expenditure (e.g., product purchase), perceived resource expenditure (e.g., willingness to pay), and satisfaction (e.g., purchase intention). Smart retail settings give consciousness composed of fantasy, imagery, and creative play to the entertainment customers and give cognition to the demanding customers composed of direct interaction, secondary source information, and intentional belief through customer experience enhancement at the organization layer. These two enable the customer move from the organism layer to the response layer. Figure 3 summarizes the customer shopping experience through applications applied to Company K's smart store.

5. Components of Smart Stores

5.1. Smart CCTV. Company K has installed 4~6 smart CCTVs per store in 3 branded flagship stores to implement personalized marketing by analyzing user behavior in store through indoor positioning and facial recognition technology and implement store layout optimization. Smart CCTV performs the four following functions: people counting, facial recognition, customer movement analysis, and interest area

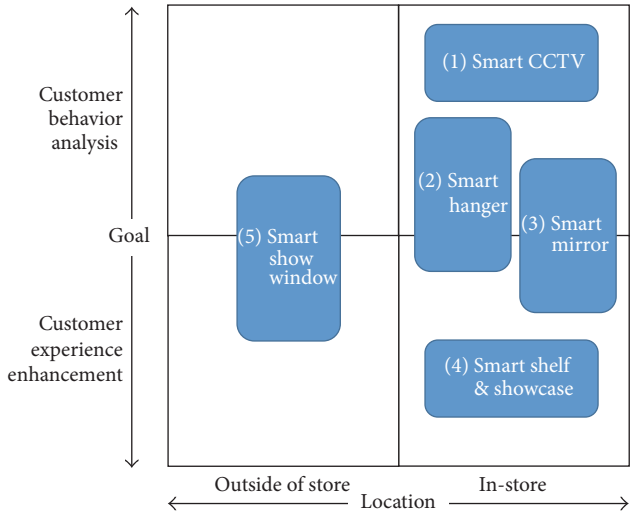


Figure 2: Solutions for K's smart stores.

analysis. First, Company K installed CCTV with sensors to measure the number of visitors at all outlets of the store. By measuring the number of customers who entered the store, it is possible to measure the purchase conversion rate, which indicates the number of visitors per day and per hour, average residence time of customers, and number of buyers compared to visitors. Therefore, it is possible to plan an efficient operation of employees. Secondly, a facial recognition sensor was attached to the CCTV installed at the main entrance of the store. As a result, it became possible to measure the number of unique visitors by solving duplicate visitors. Thirdly, the company installed CCTV for each major area of the store, and logged customers' travel route and residence time in the store. By analyzing customer shopping lines, it is made possible to improve the store layout, to increase the efficiency of the use of the store area, and to relocate the products in the store, thereby inducing the strategic purchase of products. Fourthly, Company K measured the residence time at a certain point through CCTV and introduced a system to send a notification message to the shop staff when customers stay at a specific spot for more than 30 seconds. Customers are prevented from leaving due to lack of proper guidance even though they are interested in the product, and the store personnel can understand the customer's interest area and respond to the customers rapidly.

As shown in Figure 4, it becomes possible to evaluate the movement of people, the value of the store area, and the attractiveness of the goods objectively, whereas previously this was only identified subjectively by store personnel. Furthermore, effects of promotions and events can be measured and utilized to encourage the purchases of the viewer.

The details of the technology applied for smart CCTV are as follows: firstly, sensor cameras are installed on the ceiling of all entrances in the facility to count customers. Then a counting area is set inside the entrance, so that only those who are out of the area are counted. Secondly, a facial recognition system is installed through five sensor cameras on the ceiling of the main entrance in order to get five face images of the customer through the processor. The supporting hardware consists of a camera module, a twist unit (TWU), and a processor. TWU converts video signals into differential signals and the process encrypts and transmits upper body photographs to a management PC. In order to protect the privacy of the customer, the management PC extracts the attributes for the face recognition from the pictures and deletes the photographs. The software of GikenTrastem provides the gender and the age resolution. The test results showed an accuracy of about 80%. However, the quality of recognition accuracy depends on the mask or face direction. Thirdly, the analysis of the customer movement and the analysis of the interest area are performed by measuring the travelling time of the customer through the sensor installed on the ceiling and the time spent at the specific location and time recording. In particular, Company K has introduced a system that notifies customers of the products they are interested in by sending signals to smart pad (iPad) and wearable device (iWatch).

5.2. Smart Hanger. Company K has installed a smart hanger in the flagship store of S and L brands to provide an intuitive user experience by using beacon and digital display as shown in Figure 5. Smart hangers analyze products of interest, customer experience, and customer response. The implementation method and effect are as follows. Firstly, Company K can attach a beacon with an accelerometer on the smart hanger to measure how many customers were holding the hanger with the item and what distance they have moved with the hanger. The company can link the logs made by the smart hanger with the purchase data and analyze the link between the goods of interest and the purchase items. This is similar to linking e-commerce to click-through pages, shopping carts, and purchase items. Through this, Company K has been able to target marketing by tracking and analyzing the processes from the point of customer's attention to the product to the process of purchasing. Secondly, Company K has smart hangers in conjunction with digital signage to provide customers with a new store experience. The display in standby mode shows the top three items and promotional images, such as model photos. However, if a customer holds a hanger with a beacon, the media pole digital display on the top of the hanger shows details of the product such as material and price and the number of customers who previously selected the product. In addition, when the customer takes the coat hanger to the mirror, details of the product are displayed in the mirror. This allows the customer to learn detailed information about the product without help of the store staff and make purchasing decisions based on the purchasing tendencies of other customers. Thirdly, smart hangers can also help retailers gather customer data by measuring and analyzing in-store behavior. The beacon embedded in smart hangers also enables smart pads and wearable devices used by the store staff to send messages related to products of customer interest, so that they can be prepared for invisible customer interaction.

Smart hangers consists of a hanger with an accelerometer-installed beacon, a 3-sided media pole, a video wall, a digital display, and an Android set-top box. Here, the accelerometer

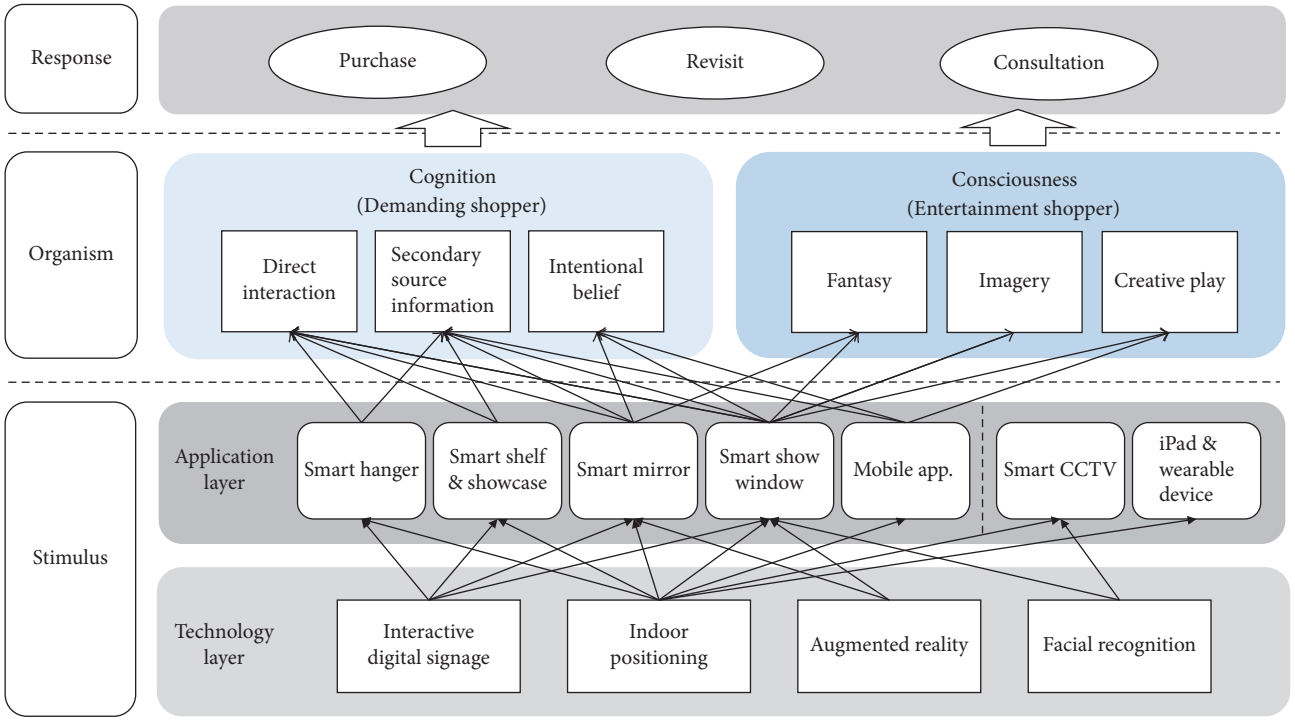


Figure 3: Application of the SOR framework.

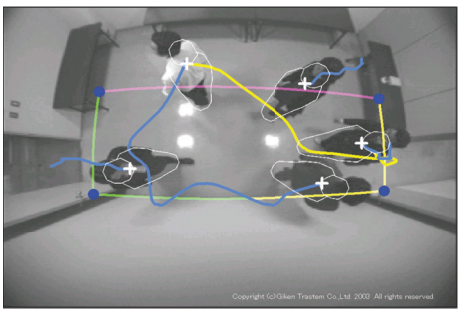


Figure 4: Operation of people counting sensor.

Figure 5: Smart hanger.

measures the time the customer picks up the hangers and movement of the hangers. The beacon attached to the smart hanger sends the signal to the digital display when the customer picks up the hanger to display detailed information about the product. The beacon of the smart hanger also sends

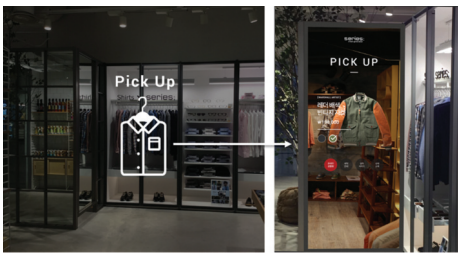


Figure 6: Smart mirror using augmented reality.

a signal to the mirror to display the details of the product when the two are in close proximity. While this is happening, the store employee receives an alarm through a smart pad and wearable devices when the customer picks up the hanger. Customers can also receive detailed information on products related to the products they are interested in and a list of recommended products through the smart pad.

5.3. Smart Mirror. Company K has been providing virtual fitting and product recommendation services to customers by combining augmented reality and interactive display technology in mirrors installed in S, L, and C brands' flagship stores (Figure 6). Through this, Company K is providing new customer experiences to customers who visit stores and improving the purchase rate by recommending products optimized for customers.

The smart mirror provides beacon-related product information, virtual fitting, SNS sharing, on-offline shopping connection, and product recommendation. Firstly, the smart

Figure 7: AR mannequin on the smart show window.

Figure 8: Show window brochure.

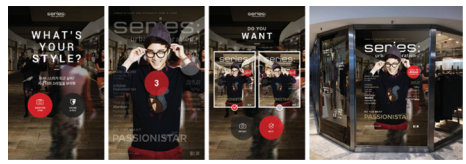


Figure 9: Facial recognition using Kinect.

mirror displays details of selected items in conjunction with the smart hanger, enabling customers to shop more proactively without the help of clerks. When the customer holds the smart hanger and goes near the smart mirror, the sensor attached to the mirror recognizes it and displays the thumbnail image of the product in the mirror display. When the customer selects the thumbnail image, the smart mirror displays the related item in the mirror display along with the detailed information, such as the product name, price, size, product review, laundry guide, and material. Customers can also use the staff call feature in the mirror display to ask additional questions. Secondly, the smart mirror provides a virtual fitting function that shoots moving images at 360 degrees and changes the color of the video after shooting to estimate the wearing shot. This allows customers to see which color best suits their needs without having to try on individual items of clothing. Thirdly, the smart mirror stores the pictures and images it takes. From there, they can be transferred using MMS and QR codes to their mobile phones, transmit them to their friends, and provide the function of allowing their friends or SNS to evaluate their choices. This not only provides new shopping experiences to customers, but also increases the rate of conversion of purchases by receiving real time ratings from friends. Fourthly, besides the smart hanger, customers can tag product barcodes in the store with barcode sensors to access detailed information about products by tagging them on the smart mirrors. In addition, the customer can put the retrieved items in their shopping cart on the online shopping malls for stores. The smart mirror implements the Omnichannel marketing function, which is merely in its initial form but also able to extend to showrooming. For this purpose, Company K has installed a mirror display with a camera, a beacon sensor, and a bar code recognition sensor on the mirror display. We have also introduced software that includes an augmented reality function to provide virtual fitting. Meanwhile, Company K has developed a mobile application for online shopping interactions, so that customers can add items viewed offline to their shopping cart on the online shopping malls for stores.

5.4. Smart Show Window. As shown in Figures 7 and 8, Company K has installed a smart interactive display in the show window outside stores for drawing customer attention through customer experience enhancement. Brand S has installed a show window with facial recognition and augmented reality technology, and brand L has installed a show window with a brochure function. Company K has developed an AR mannequin on the show window of brand S, which was built with a transparent display with the dual purpose of shop window. In order to maximize the effect of the transparent display device, the actual mannequin was also displayed inside. This mannequin is displayed in a virtual outfit chosen by the shop staff. Normally, the transparent display above the outfit on the mannequin displays information about the outfit, but the new technology allows the user to view detailed information about the outfit when they touch a button on the display. The user can also change the color of the garment, allowing them to experience an interactive display that uses augmented reality technology. In addition, shop staff can change the outfit worn on the mannequin using the iPad, meaning the display can be changed from time to time to suit seasonal fashion trends. By contrast, brand L's show window brochure has added an interactive display function to the show window display of the store that shows the contents of the store as an interactive fashion pictorial. The store can use this to contain more displays on a limited show window screen and allows the user to choose a display based on their preferences. This type of display attracts potential customers' attentions, encouraging them to linger outside the store for longer, as well as allowing customers to make decisions before entering the store and therefore increasing the ratio of purchase rate compared to visiting customers. In addition, this encourages customer engagement by enabling customers to capture photos and utilize them as pictorial models.

Company K uses facial recognition technology to measure the age range of customers and provides customized product recommendation services. At the same time, Company K uses the gamification function to attract customers who pass by and utilizes them for customer acquisition. As shown in Figure 9, Company K has installed a Kinect sensor with a facial recognition function in the mirror-type show window display of the store. This show window works like a mirror, but the Kinect sensor is activated when a potential user comes within 2 m. There are camera and proximity sensors installed behind the mirror glass which is invisible to the user. In reality, facial recognition is made possible

by linking cameras in digital information display (DID) with NEC's "Field Analyst for Signage" solution. Using this process, it is possible to measure up to 20 passengers passing at a common walking speed at the same time. Each customer's age and gender are estimated through a process consisting of three combined modules: face detection, feature extraction, and face recognition. First, we used a geometry feature-based approach to analyze the features of eyes and nose and used the geometrical relationships between them, using graph matching algorithms, for face detection. Second, we extracted image (wrinkle) and shape (geometry) features of each face image for feature extraction. Finally, we used a support vector machine (SVM) classifier in the face recognition process to improve the accuracy of gender and age estimation systems using face images. We used a single linear SVM classifier trained using the feature vector representations for gender estimation. Because gender estimation is a binary classification that distinguishes between male and female and it is easy to obtain many learning data with accurate gender information, the SVM classifier achieves a high recognition rate of over 95%. On the other hand, it is difficult to improve accuracy in age estimation, because it is difficult to obtain learning data with accurate age information and there is no standard age estimation method. Therefore, we performed multilabel age classification and then performed one-versus-one linear SVM arrangement for each age group. Furthermore, it has also become possible to classify the floating population passing through the store and the customers visiting the store by gender (male/female) and age (1-year-old).

This makes it possible to measure the effect of the show window display by measuring the residence time (display residence time) in front of the show window, the time of the gaze (screen gaze time), and the distance (distance between the screen and the person). When the upper illuminance is above 500 lux, the accuracy is more than 90% for flow population, 80% for sex, and 70~75% for age group. However, since privacy concerns exist in facial recognition [64], images and personal information are not stored in accordance with Article 25 (1) of the Personal Data Protection Act for the protection of personal information. Instead, data is stored as raw numeric and text data used for customer analysis. Stores use facial recognition technology to enhance the customer's interest by providing appropriate games and provide product recommendation services based on information obtained. The customer can participate in a game (age estimation game) that calculates their age through the display installed in the store show window. The photographs taken here are used for customer segmentation by utilizing facial recognition technology and provides different product recommendation services appropriate to age and gender based on face recognition results through the smart window mirror display.

5.5. Smart Shelf and Showcase. As shown in Figures 10 and 11, Company K has installed a smart shelf and smart showcase that combines indoor positioning and smart interactive display technology in brick-and-mortar stores for increased customer retention through customer experience enhancement. Tags on fashion accessories, such as handbags, have beacons attached to accelerometers attached to them. The operation of these tags are similar to smart hangers. When the user picks up a fashion accessory, the sensor attached to the wall display captures the movement of the beacon and displays product information. If several people pick up a product at the same time, this information is displayed on the mirror near to the product. It also provides an Omnichannel shopping function that links offline shopping items to online shopping carts in conjunction with online shopping malls. At the same time, brand C from Company K offers a new user experience by combining an interactive display with a showcase, which displays accessories such as wallets. Technically, it combines touch screen-based digital signage with existing showcases to attract visitors' attention and allows customers to view detailed information about the products they are interested in without being directly involved. The smart showcase provides detailed information about the selected product such as material, release date, selling price, and related product information when the product shown on the display is touched or the attached accessory is placed on the RFID reader. This enables customers who do not feel comfortable asking store staff to inquire about the product without direct interaction. In addition, store staff are able to respond better to customers' needs through receiving customers' inquiry history on their iPad.

Figure 10: Smart shelf using interactive display.

Figure 11: Smart showcase using interactive display.

5.6. Smart Pad and Wearable Device. Company K utilizes information about customer preferences accumulated in the store and provides it to store staff who use smart pads (iPad) and wearable devices (iWatch) for efficient customer response. In addition, a dedicated application was developed to process data collected from various sensors and to provide customer information to shop staff. This enables shop staff to improve customers' experiences in relation to customer identification, product recommendation, and store management. Firstly, store staff can focus on their customers and identify their needs without having to ask questions. Customer information collected from smart CCTV's customer entry and point of interest notifications, smart hanger, and smart

merchandise is transferred to shop staff through smart pads and wearable devices. This allows shop staff to understand customers' interest areas and which products the customer are interested in. Secondly, analyzing integrated customer behavior data allows for efficient product recommendation. When a customer hesitates to purchase a product, the recommended product and the related product list on the smart pad may prevent the customer from leaving the store and provide functions that encourage additional purchases. In addition, the availability of information such as color, sales ranking, and inventory information on smart pads make it possible for store staff to provide quick response to the customers. Thirdly, the store staff can identify the location of the items in the store and the inventory quantity as well as a customer's staying time and list of popular products through smart pad, thereby enabling efficient store management. In addition, shop staff can use the smart pad to change the display of the augmented reality mannequin displayed in the smart window, or to change the location of items on the smart showcase. Smart pads and wearable devices provided to employees in stores are used to provide real time trigger to customers by gathering information from smart CCTV, smart hangers, and smart displays and processing through event processing and data analysis. In other words, smart pads and wearable devices are important solutions to improve customer satisfaction and offer relevancy in smart stores.

5.7. Smartphone Application. Company K conducts targeted marketing through indoor positioning technology and smartphone application using beacons. The mobile application developed for this purpose provides push type information delivery, on-offline shopping link, SNS support, membership, and so forth. The following are detailed functions: firstly, it is possible to implement location-based marketing for retailers by linking indoor positioning technology and mobile application. When a customer who has installed a smart application on their smartphone passes a store, the customer can use information, such as purchase patterns, product details, events, and coupons through smartphone application. Secondly, showrooming and web rooming, which combine online mobile shopping and actual in-store shopping, becomes possible. Customers can find products of interest in offline stores and then add them into the shopping cart for mobile shopping using the smartphone application. In addition, when a customer registers in a smartphone application or comes online, shops can display location information through the mobile application when customers search for offline stores. Thirdly, the smart application provides a new shopping experience by linking the application with SNS services. Customers can transfer images shot through smart mirrors to their friends to receive their opinions and can promote products to their friends by putting products into the shopping cart. Finally, the membership function in the mobile application supports customized shopping. If a customer submits their membership card barcode through the smart window or the smart mirror, they can receive recommendations based on their interest and can receive convenient shopping service based on their purchase history.

6. Discussions

The development of information and communication technologies is fundamentally changing the entire industry, including the retail industry [65–67]. The technologies related to Internet of Everything (IoE) connect people, processes, data, and things in the world, and add new value by adding information to things. Company K's Smart store consists of various Internet of things technologies, such as smart hanger based on sensor technologies, smart CCTV based on indoor positioning and facial recognition, and smart mirror based on augmented reality and digital signage. The term "Internet of Everything" is an extension of the Internet of things. Dave Evans, who originated the concept of IoE, defined IoE as "the intelligent connection of people, process, data and things". IoT focuses on machine-to-machine (M2M) communications, which are communications between machines, while the more expansive IoE concept includes M2M communications, machine-to-people (M2P), and people-to-people (P2P) interactions. We cited the term IoE because the technologies used in the smart store include M2P and P2P communications as well as M2M.

In this environment, companies must utilize big data, including user behaviors, to optimize operations, adjust prices, predict demand, and provide desired product configurations. In this regard, Gartner forecasts that customer digital assistants will recognize individuals by face and voice across channels and partners by the end of 2018 [68]. They argued that customer experiences with conventional offline stores and online shopping malls will be integrated into a multichannel retail setting and this new service will support tech-savvy customers. Customers' increased willingness to adopt facial and voice recognition technologies allows firms to acquire large amounts of information related to the consumer's purchase process. The "smart store" is an example of the paradigm shift from "businesses use technology" to "technology defines businesses" in relation to business and technology. Recently, the characteristics of the retail industry have been transformed to "smart store" due to smart retail settings that utilize advanced technology.

6.1. Theoretical Contributions. Our research provides the following theoretical contributions. Firstly, we conducted an integrated research that combines marketing and computing perspectives in terms of the purpose and means of implementing a smart store. A variety of previous studies have addressed the technical elements and analysis methods for implementing customer recognition [50, 53, 69], indoor location analysis [5, 13], customer behavior logs analysis [18], and augmented realty [70–72].

Secondly, our case includes case studies of indoor positioning, augmented reality, facial recognition, and interactive digital signage. Our case incorporates various technical elements and IoT into various objects in fashion shops, such as hanger, fitting mirror, show window, and shelf. While most previous studies have put emphasis on the technical content of one or two components of the smart store, our study is a comprehensive case study. In particular, our case has actually

Table 4: Comparison of sales before and after installation of smart store.

	Before introduction (Average sales per store within brand)	After introduction (Average sales per store within brand)	Increase/decrease width	Fluctuation rate	Installation time
S brand	108.90%	141.30%	32.40%	29.76%	July, 2015
L brand	45.98%	61.56%	15.58%	33.87%	December, 2015
C brand	26.19%	34.13%	7.94%	30.33%	December, 2015

built in three branded flagship stores of the leading fashion company in Korea.

Thirdly, we looked at a case of collecting data that can analyze customer behavior in the conventional offline store using various IoT technologies. This allows us to offer personalized marketing proposals to our users and optimize their circulation and layout. Tracking the user's location, identifying products of interest, and tracking the user's gender and age are similar to the analytic process of estimating the user's personalities and preferences through online web log analysis. In the past, it was almost impossible to collect and analyze data on customer preferences in an offline environment. However, this study shows that it is becoming possible to analyze customer behavior using sensors and IoT technology installed in the retail stores.

Fourthly, this study has demonstrated that retail smart settings in brick-and-mortar stores can increase sale within a short period. To measure the economic performance of Company K's smart store, we compared the sales results between the "smart store" and other stores in the same brand before and after introducing "smart store." This approach is reasonable because the fashion industry has highly fluctuating sales due to new product purchases, promotions, and seasonal changes. Table 4 shows the ratio of the store sales to the average sales of all stores in the brand as a comparison index. In the case of brand S, sales of the store increased from 108.90% to 141.30% after introducing the smart store. Brand L's sales increased from 45.98% to 61.56%, and brand C's sales increased from 26.19% to 34.13%. Each brand showed a similar sales increase of around 30%. Average sales of brand L and C are lower than the average sales growth because there are many stores in department stores with high sales per store.

6.2. Limitations and Future Researches. Firstly, cost is the biggest problem with smart retail settings in brick-and-mortar stores. All retailers aim for larger profit. However, smart store technology is still too expensive to use in retail settings. A low return of investment caused by excessive investment costs and a long payback period are obstacles in spreading the smart store. For this reason, the fashion industry is building smart stores mainly in flagship stores. Therefore, it is important to overcome this problem in order to analyze user behavior at a lower cost.

Secondly, from a long-term perspective, it is necessary to examine whether sales increase, increase of visitors, and purchase conversion rate were by introducing smart store. In addition, the technology and equipment required for the smart store are often expensive, so it is necessary to research whether the increase in profits from introducing the smart store exceeds the cost required. In the future, if a brand is recognized through the introduction of a flagship store, and if it has played a role in attracting customers' attention, research on the efficiency of technology introduction is needed for universal utilization in future brick-and-mortar stores. In addition, seasonal factors, promotional effects, and longer-term changes in earnings should be considered to account for changes in sales across multiple shopping malls.

Thirdly, the use of personal information, such as location, appearance, and preference of consumers collected for providing new experiences, immersions, and customized proposals comes with the risk of privacy infringement [73, 74]. Therefore, it is necessary to conduct detailed studies on the interest conflict issues between the efficiency improvement of using personal information and the protection of privacy, including legal and ethical reviews on the privacy concern.

7. Conclusion

In this paper, we investigated persuasive marketing and immersive customer experience based on various Internet of things technologies through a case study on smart store in a Korean fashion company. Smart retail settings will change customer behaviors through customer experience enhancement in the marketing perspective. In addition, various IoT technologies such as indoor positioning, augmented reality, facial recognition, and interactive display make it possible to create solutions and components for smart store implementations in the computing perspective. Although this paper thoroughly surveys the state-of-the-art marketing practices and technical applications in smart store, we believe that there should be further practical and theoretical research on smart store to implement it successfully in various situations.

Competing Interests

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

Acknowledgments

This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2015S1A5A8016824).

References

[1] eMarketer, "Retail Sales Worldwide Will Top #2 Trillion This Year," March 9, 2016 http://www.emarketer.com/Article/Retail-Sales-Worldwide-Will-Top-22-Trillion-This-Year/1011765#sthash.22DpGkTB.dpuf.

[2] pwc, "Total Retail 2016: They say they want a revolution," pwc, 2016.

[3] A. M. Fiore and J. Kim, "An integrative framework capturing experiential and utilitarian shopping experience," *International Journal of Retail & Distribution Management*, vol. 35, no. 6, pp. 421–442, 2007.

[4] Z. Jiang, J. Chan, B. C. Y. Tan, and W. S. Chua, "Effects of interactivity on website involvement and purchase intention," *Journal of the Association of Information Systems*, vol. 11, no. 1, pp. 34–59, 2010.

[5] A. Yaeli, P. Bak, G. Feigenblat et al., "Understanding customer behavior using indoor location analysis and visualization," *IBM Journal of Research and Development*, vol. 58, no. 5/6, pp. 3:1–3:12, 2014.

[6] S. K. Hui, P. S. Fader, and E. T. Bradlow, "Path data in marketing: an integrative framework and prospectus for model building," *Marketing Science*, vol. 28, no. 2, pp. 320–335, 2009.

[7] Z. Xiaoling, S. Li, R. R. Burke, and A. Leykin, "An examination of social influence on shopper behavior using video tracking data," *Journal of Marketing*, vol. 78, no. 5, pp. 24–41, 2014.

[8] G. Petro, "The future of fashion retailing, revisited: part 2—Zara," *Forbes*, 2015.

[9] MarketsandMarkets, *Indoor Location Market by Solution (Tag-Based, RF-Based, Sensor-Based), by Application (Indoor Maps & Navigation, Indoor Location-based Analytics, Tracking & Tracing, Monitoring & Emergency Management), by Service, by Vertical, & by Region—Global Forecast Up to 2019*, 2014.

[10] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: indoor localization via channel response," *ACM Computing Surveys*, vol. 46, no. 2, article 25, 2013.

[11] N. Fallah, I. Apostolopoulos, K. Bekris, and E. Folmer, "Indoor human navigation systems: a survey," *Interacting with Computers*, vol. 25, no. 1, pp. 21–33, 2013.

[12] X. Li and N. Alsindi, "Recent advances in indoor geolocation techniques," *International Journal of Wireless Information Networks*, vol. 20, no. 4, pp. 243–245, 2013.

[13] K. Yada, "String analysis technique for shopping path in a supermarket," *Journal of Intelligent Information Systems*, vol. 36, no. 3, pp. 385–402, 2011.

[14] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007.

[15] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.

[16] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1281–1293, 2013.

[17] K. Takai and K. Yada, "Relation between stay-time and purchase probability based on RFID data in a Japanese supermarket," in *Knowledge-Based and Intelligent Information and Engineering Systems*, pp. 254–263, Springer, Berlin, Germany, 2010.

[18] A. Ogino, T. Kobayashi, Y. Iida, and T. Kato, "Smart store understanding consumer's preference through behavior logs," in *Internationalization, Design and Global Development*, pp. 385–392, Springer, 2011.

[19] G. Naylor, S. B. Kleiser, J. Baker, and E. Yorkston, "Using transformational appeals to enhance the retail experience," *Journal of Retailing*, vol. 84, no. 1, pp. 49–57, 2008.

[20] N. M. Puccinelli, R. C. Goodstein, D. Grewal, R. Price, P. Raghubir, and D. Stewart, "Customer experience management in retailing: understanding the buying process," *Journal of Retailing*, vol. 85, no. 1, pp. 15–30, 2009.

[21] P. C. Verhoef, K. N. Lemon, A. Parasuraman, A. Roggeveen, M. Tsiros, and L. A. Schlesinger, "Customer experience creation: determinants, dynamics and management strategies," *Journal of Retailing*, vol. 85, no. 1, pp. 31–41, 2009.

[22] C. Meyer and A. Schwager, "Understanding customer experience," *Harvard Business Review*, vol. 85, no. 2, pp. 116–157, 2007.

[23] J. Möller and S. Herm, "Shaping retail brand personality perceptions by bodily experiences," *Journal of Retailing*, vol. 89, no. 4, pp. 438–446, 2013.

[24] L. Esbjerg, B. B. Jensen, T. Bech-Larsen, M. D. De Barcellos, Y. Boztug, and K. G. Grunert, "An integrative conceptual framework for analyzing customer satisfaction with shopping trip experiences in grocery retailing," *Journal of Retailing and Consumer Services*, vol. 19, no. 4, pp. 445–456, 2012.

[25] A. Parasuraman, "Technology Readiness Index (TRI) a multiple-item scale to measure readiness to embrace new technologies," *Journal of Service Research*, vol. 2, no. 4, pp. 307–320, 2000.

[26] M. L. Meuter, M. J. Bitner, A. L. Ostrom, and S. W. Brown, "Choosing among alternative service delivery modes: an investigation of customer trial of self-service technologies," *Journal of Marketing*, vol. 69, no. 2, pp. 61–83, 2005.

[27] J. M. Curran, M. L. Meuter, and C. F. Surprenant, "Intentions to use self-service technologies: a confluence of multiple attitudes," *Journal of Service Research*, vol. 5, no. 3, pp. 209–224, 2003.

[28] P. A. Dabholkar and R. P. Bagozzi, "An attitudinal model of technology-based self-service: moderating effects of consumer traits and situational factors," *Journal of the Academy of Marketing Science*, vol. 30, no. 3, pp. 184–201, 2002.

[29] M. L. Meuter, A. L. Ostrom, R. I. Roundtree, and M. J. Bitner, "Self-service technologies: understanding customer satisfaction with technology-based service encounters," *Journal of Marketing*, vol. 64, no. 3, pp. 50–64, 2000.

[30] B. Weijters, D. Rangarajan, T. Falk, and N. Schillewaert, "Determinants and outcomes of customers' use of self-service technology in a retail setting," *Journal of Service Research*, vol. 10, no. 1, pp. 3–21, 2007.

[31] B. Schmitt, *Experiential Marketing: How to Get Customers to Sense, Feel, Think, Act, Relate*, Simon and Schuster, New York, NY, USA, 2000.

[32] K. Curran, E. Furey, T. Lunney, J. Santos, D. Woods, and A. McCaughey, "An evaluation of indoor location determination technologies," *Journal of Location Based Services*, vol. 5, no. 2, pp. 61–78, 2011.

[33] M. Koohikamali, N. Gerhart, and M. Mousavizadeh, "Location disclosure on LB-SNAs: the role of incentives on sharing behavior," *Decision Support Systems*, vol. 71, pp. 78–87, 2015.

[34] K. Li and T. C. Du, "Building a targeted mobile advertising system for location-based services," *Decision Support Systems*, vol. 54, no. 1, pp. 1–8, 2012.

[35] M. Popa, L. Rothkrantz, Z. Yang, P. Wiggers, R. Braspenning, and C. Shan, "Analysis of shopping behavior based on surveillance system," in *Proceedings of the 2010 IEEE International*

Conference on Systems, Man and Cybernetics (SMC '10), pp. 2512–2519, October 2010.

[36] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.

[37] F.-C. Cheng, S.-C. Huang, and S.-J. Ruan, "Scene analysis for object detection in advanced surveillance systems using laplacian distribution model," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 41, no. 5, pp. 589–598, 2011.

[38] S. B. Contributor, "The future of retail: 4 trends bringing the virtual experience into reality [infographic]," *Forbes*, 2015.

[39] Y.-S. Chiou, C.-L. Wang, and S.-C. Yeh, "An adaptive location estimator using tracking algorithms for indoor WLANs," *Wireless Networks*, vol. 16, no. 7, pp. 1987–2012, 2010.

[40] P. Daponte, L. De Vito, F. Picariello, and M. Riccio, "State of the art and future developments of the Augmented Reality for measurement applications," *Measurement*, vol. 57, pp. 53–70, 2014.

[41] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.

[42] A. State, G. Hirota, D. T. Chen, W. F. Garrett, and M. A. Livingston, "Superior augmented reality registration by integrating landmark tracking and magnetic tracking," in *Proceedings of the Computer Graphics Conference (SIGGRAPH '96)*, pp. 429–438, August 1996.

[43] M. Bajura and U. Neumann, "Dynamic registration correction in augmented-reality systems," in *Proceedings of the IEEE Annual Virtual Reality International Symposium*, pp. 189–196, Research Triangle Park, NC, USA, March 1995.

[44] A. S. M. Mahfujur Rahman, T. T. Tran, S. A. Hossain, and A. El Saddik, "Augmented rendering of makeup features in a smart interactive mirror system for decision support in cosmetic products selection," in *Proceedings of the IEEE/ACM 14th International Symposium on Distributed Simulation and Real Time Applications (DS-RT '10)*, pp. 203–206, Fairfax, Va, USA, October 2010.

[45] T. Nakajima and V. Lehdonvirta, "Designing motivation using persuasive ambient mirrors," *Personal and Ubiquitous Computing*, vol. 17, no. 1, pp. 107–126, 2013.

[46] K. Higuchi, S. Sea-Ueng, Y. Watanabe et al., "Modeling KANSEI through real world interaction with ubiquitous information environment—smart sphere and smart store," in *Proceedings of the 6th Asia Design Conference*, Tsukuba, Japan, October 2003.

[47] S. Longo, E. Kovacs, J. Franke, and M. Martin, "Enriching shopping experiences with pervasive displays and smart things," in *Proceedings of the ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, pp. 991–998, Zurich, Switzerland, September 2013.

[48] A. Bodhani, "Shops offer the e-tail experience," *Engineering and Technology*, vol. 7, no. 5, pp. 46–49, 2012.

[49] A. Farshidi, "The new retail experience and its unaddressed privacy concerns: how RFID and mobile location analytics are collecting customer information," *Journal of Law, Technology, & the Internet*, vol. 7, no. 1, article 15, 2016.

[50] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: a literature survey," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[51] G. Anthes, "Deep learning comes of age," *Communications of the ACM*, vol. 56, no. 6, pp. 13–15, 2013.

[52] R. Brunelli and T. Poggio, "Face recognition: features versus templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042–1052, 1993.

[53] H. Zhou, A. Mian, L. Wei, D. Creighton, M. Hossny, and S. Nahavandi, "Recent advances on singlemodal and multimodal face recognition: a survey," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 6, pp. 701–716, 2014.

[54] A. Azazi, S. Lebai Lutfi, I. Venkat, and F. Fernández-Martínez, "Towards a robust affect recognition: automatic facial expression recognition in 3D faces," *Expert Systems with Applications*, vol. 42, no. 6, pp. 3056–3066, 2015.

[55] Z. Guo, Y.-N. Zhang, Y. Xia, Z.-G. Lin, Y.-Y. Fan, and D. D. Feng, "Multi-pose 3D face recognition based on 2D sparse representation," *Journal of Visual Communication and Image Representation*, vol. 24, no. 2, pp. 117–126, 2013.

[56] D. A. Socolinsky and A. Selinger, "Thermal face recognition in an operational scenario," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. II-1012–II-1019, IEEE, Washington, DC, USA, July 2004.

[57] B. Kidwell, R. G. McFarland, and R. A. Avila, "Perceiving emotion in the buyer-seller interchange: the moderated impact on performance," *Journal of Personal Selling and Sales Management*, vol. 27, no. 2, pp. 119–132, 2007.

[58] F. Qiao and W. G. Griffin, "A content analysis of experimental technologies in award-winning creative strategies," *Journal of Interactive Advertising*, vol. 16, no. 2, pp. 145–156, 2016.

[59] A. L. Roggeveen, J. Nordfält, and D. Grewal, "Do digital displays enhance sales? Role of retail format and message content," *Journal of Retailing*, vol. 92, no. 1, pp. 122–131, 2016.

[60] S. D. Guler, M. Gannon, and K. Sicchio, "Speculations on wearable futures," in *Crafting Wearables*, pp. 183–195, Springer, Berlin, Germany, 2016.

[61] A. Kent, M. Vianello, M. B. Cano, and E. Helberger, "Omnichannel fashion retail and channel integration: the case of department stores," in *Handbook of Research on Global Fashion Management and Merchandising*, Advances in Logistics, Operations, and Management Science, pp. 398–419, IGI Global, Hershey, Pa, USA, 2016.

[62] D. Pederzoli, "ICT and retail: state of the art and prospects," in *Information and Communication Technologies in Organizations and Society*, pp. 329–336, Springer, Berlin, Germany, 2016.

[63] J. A. Russell and A. Mehrabian, *An Approach to Environmental Psychology*, MIT Press, Cambridge, UK, 1974.

[64] E. Klarreich, "Hello, my name is…," *Communications of the ACM*, vol. 57, no. 8, pp. 17–19, 2014.

[65] E. Pantano, "Innovation management in retailing: from consumer perspective to corporate strategy," *Journal of Retailing and Consumer Services*, vol. 21, no. 5, pp. 825–826, 2014.

[66] E. Pantano and A. Tavernise, "Learning cultural heritage through information and communication technologies: a case study," in *Learning Culture and Language through ICTs: Methods for Enhanced Instruction*, pp. 103–119, IGI Global, Hershey, Pa, USA, 2009.

[67] E. Pantano, "New technologies and retailing: trends and directions," *Journal of Retailing and Consumer Services*, vol. 17, no. 3, pp. 171–172, 2010.

[68] V. Woods, *Gartner Reveals Top Predictions for IT Organizations and Users for 2016 and Beyond*, Gartner Inc, 2015.

[69] B. R. Abidi, N. R. Aragam, Y. Yao, and M. A. Abidi, "Survey and analysis of multimodal sensor planning and integration for wide area surveillance," *ACM Computing Surveys*, vol. 41, no. 1, article 7, 36 pages, 2008.

[70] V. G. Cerf, "Augmented reality," *Communications of the ACM*, vol. 57, no. 9, p. 7, 2014.

[71] X. Zhang, Y.-H. Yang, Z. Han, H. Wang, and C. Gao, "Object class detection: a survey," *ACM Computing Surveys*, vol. 46, no. 1, article 10, 2013.

[72] Á. Csapó and G. Wersényi, "Overview of auditory representations in human-machine interfaces," *ACM Computing Surveys*, vol. 46, no. 2, pp. 1–23, 2013.

[73] F. Roesner, T. O. Kohno, and D. Molnar, "Security and privacy for augmented reality systems," *Communications of the ACM*, vol. 57, no. 4, pp. 88–96, 2014.

[74] H. Xu, H.-H. Teo, B. C. Y. Tan, and R. Agarwal, "Effects of individual self-protection, industry self-regulation, and government regulation on privacy concerns: a study of location-based services," *Information Systems Research*, vol. 23, no. 4, pp. 1342–1363, 2012.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 3243570, 12 pages
https://doi.org/10.1155/2017/3243570

Research Article

A Secure Localization Approach Using Mutual Authentication and Insider Node Validation in Wireless Sensor Networks

Gulshan Kumar,[1] Mritunjay Kumar Rai,[2] Hye-jin Kim,[3] and Rahul Saha[4]

[1]*School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India*
[2]*School of Electronics and Communication Engineering, Lovely Professional University, Phagwara, Punjab, India*
[3]*Business Administration Research Institute, Sungshin W. University, 2 Bomun-ro 34da gil, Seongbuk-gu, Seoul, Republic of Korea*
[4]*School of Computer Applications, Lovely Professional University, Phagwara, Punjab, India*

Correspondence should be addressed to Hye-jin Kim; hyejinaa@daum.net

Received 20 September 2016; Accepted 17 November 2016; Published 26 February 2017

Academic Editor: Jaegeol Yim

Localization is a concerning issue in the applications of wireless sensor networks. Along with the accuracy of the location estimation of the sensor nodes, the security of the estimation is another priority. Wireless sensor networks often face various attacks where the attackers try to manipulate the estimated location or try to provide false beacons. In this paper, we have proposed a methodology that will address this problem of security aspects in localization of the sensor nodes. Moreover, we have considered the network environment with random node deployment and mobility as these two conditions are less addressed in previous research works. Further, our proposed algorithm provides low overhead due to the usage of less control messages in a limited transmission range. In addition, we have also proposed an algorithm to detect the malicious anchor nodes inside the network. The simulated results show that our proposed algorithm is efficient in terms of time consumption, localization accuracy, and localization ratio in the presence of malicious nodes.

1. Introduction

Localization [1, 2] defines the calculation of the location or position of sensor nodes in wireless sensor networks (WSNs). The dynamic need of the applications has made the deployment of WSNs extended from static to mobile. Such networks are dynamic and therefore the localization of nodes is also changeable and thus makes the process a critical factor in WSNs. The knowledge of the physical location of a network entity helps in different applications and services [3–5]. The main consideration of location discovery is a set of special nodes known as anchor nodes, which are resource privileged having more storage and computational capacity. Using the location of anchor nodes, other unknown nodes compute their location in different ways. Therefore, it is critical that malicious anchor nodes need to be prevented from providing false location information as the unknown nodes completely depend on the anchor nodes for computing their own location [6]. WSNs attract the adversaries in a very general way. Attacks are executed by the internal nodes as well as external nodes. Therefore, it is compulsory that the localization techniques should be secured enough [7]. The secured localization process must prevent both malicious insider nodes from misrepresenting their location and outside entities from performing intrusion with the location determination process. The security requirements for localization techniques must include privacy of the location information, authorization for legitimate nodes and the integrity to identify any kind of deviation from true location. Further, information availability to compute proper location is also required for a secured localization process. The accuracy of nodes' locations can be considered on the basis of two aspects. On one hand, nodes (anchor or unknown) need to calculate their correct position depending upon some references, which is called localization estimation (Figure 1(a)). On the other hand, the Base Station (BS) also needs to ensure that the location estimations it has received are correct. Thus, we need to verify the locations received from the nodes. This is called location verification (Figure 1(b)). In this paper, we have introduced a secured localization process using mutual

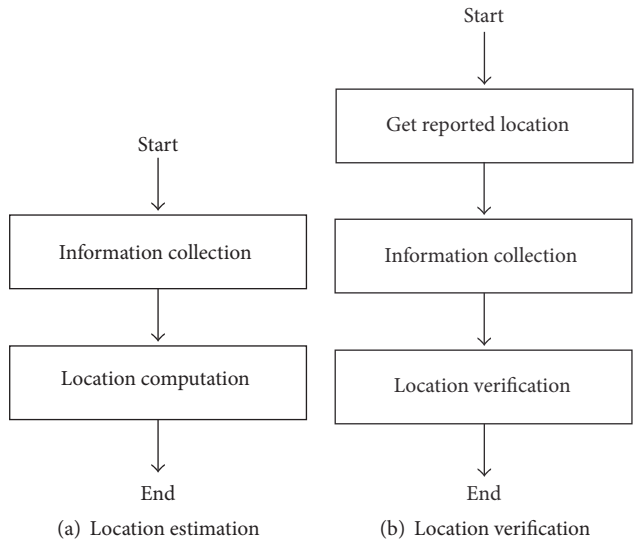


Figure 1: Localization system.

authentication and validation of insider nodes. The rest of the paper has been organized as follows. Section 2 explains the attack model, that is, the different attacks on localization systems in WSNs. Related work in this line of work has been cited in Section 3. The proposed algorithm is discussed in Section 4 along with the detailed network model and its related assumptions are in Section 5. The results of the simulation have been explained in Section 6. Finally, we have concluded the paper in Section 7.

2. Attack Model

Many attacks [8] have been studied on localization system. Attacks are executed in the information collection process in location estimation phase as well as location verification phase. There are several types of elementary and combinational attacks that can be executed in localization systems. Table 1 summarizes the layer wise attacks in WSNs localization process [9].

2.1. Elementary Attacks. Elementary attacks are the prime attacks which have their own technical aspects of execution. Some of such attacks are discussed below.

Range Change Attack. In this attack an attacker changes the range or Angle of Arrival (AoA) measurements among nodes. This attack affects both localization estimation and location verification systems. For example, reducing or increasing the range measurement between node A and node B will lead to malicious estimation of locations of B shown by green dotted circles in Figure 2.

False Beacon Location Attack. In this attack an attacker makes the victim node receive false estimated locations. For example, an attacker gains control over a beacon or anchor node and then it make the node broadcast false location. False reported location attack is generally executed in a location veri cation system where a malicious anchor node or unknown node reports false location.

2.2. Combinational Attacks. Combinational attacks are those who merge different technicalities of elementary attacks and create overall malicious affect. Some of the important combinational attacks are listed below.

Impersonation. In this attack an attacker makes its identity be as a legitimate node in the network. For example, in localization systems, an attacker spoofs the anchor nodes' identity and broadcasts false locations. This leads to erroneous range measurements. In location verification systems, an attacker impersonates a victim node to make verifiers believe that the original node is at the attacker's location.

Sybil Attack. In this attack a malicious node has the capability of presenting itself as different identities in a network to function as distinct nodes. These multiple identities are called Sybil nodes. It sends false information like position of beacon nodes and erroneous strength of signal. By masquerading and disguising as multiple identities, this type of malicious node gains control over the network.

Location-Reference Attack. This attack is executed against the localization phase. Each common node gets a location-reference set $\langle \mathrm{loc}_i, d_i \rangle$ for localization where loc_i is the location of beacon i and d_i is the distance between the beacon and the common node. In this attack the attacker makes the compromised beacons broadcast false locations and distorts the distance measurements between beacons and common nodes. The attack can be classified into three types: (a) uncoordinated attack, (b) collusion attack, and (c) pollution attack. Exemplary scenarios are shown in Figures 3(a), 3(b), and 3(c), respectively. Red nodes represent the attacker nodes,

Table 1: Summarization of layer wise attacks on localization in WSNs [9].

Layers	Attacks	Attack behaviour	Results
Physical layer	Stealing	Signal eavesdropping and tampering	Packet error and packet loss
	Jamming	Sending jamming signal in the working frequency range	Packet loss
Data link layer	Collision	Repetition of messages	Packet loss
	Exhaustion	Sending of unnecessary message	Packet loss
	Unfairness	Explicitly take the control of the channel	Packet loss
Network layer	DoS Attacks	Exhaustion of energy of the unknown nodes	Packet loss
	Selective forwarding	Selectively forward packets	Packet loss
	Sybil	Possessing multiple identities	Packet error
	Sinkhole	Maliciously tamper with routing	Packet error
	Wormhole	Shortening the distance to make a fast routing path	Packet loss
Transport layer	Flooding	Establishing false connections	Packet loss
	Tampering	Tampering localization beacons	Packet error

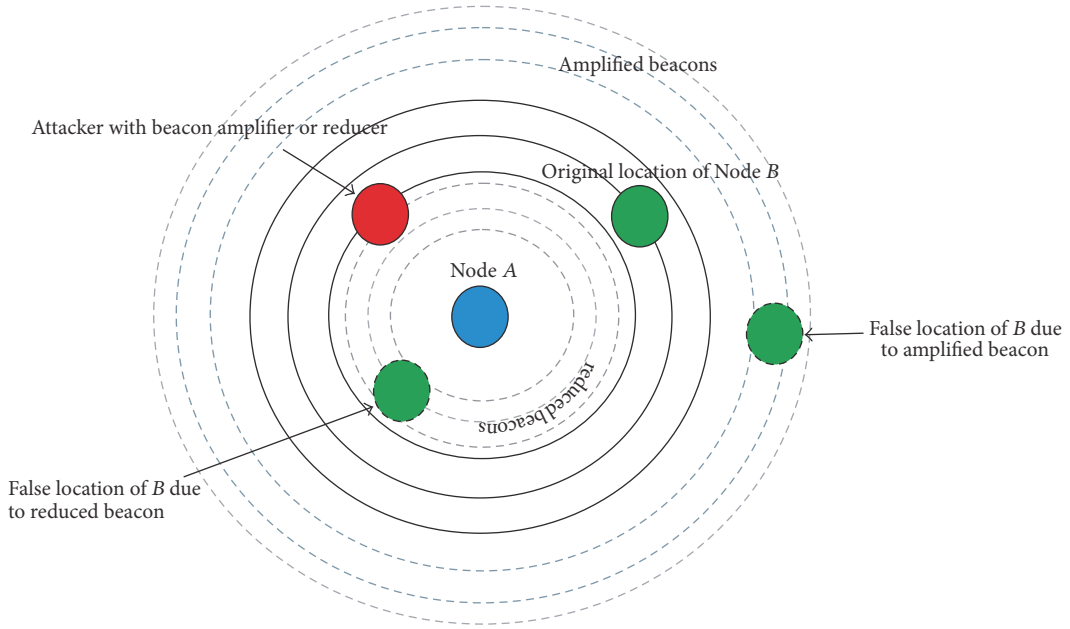


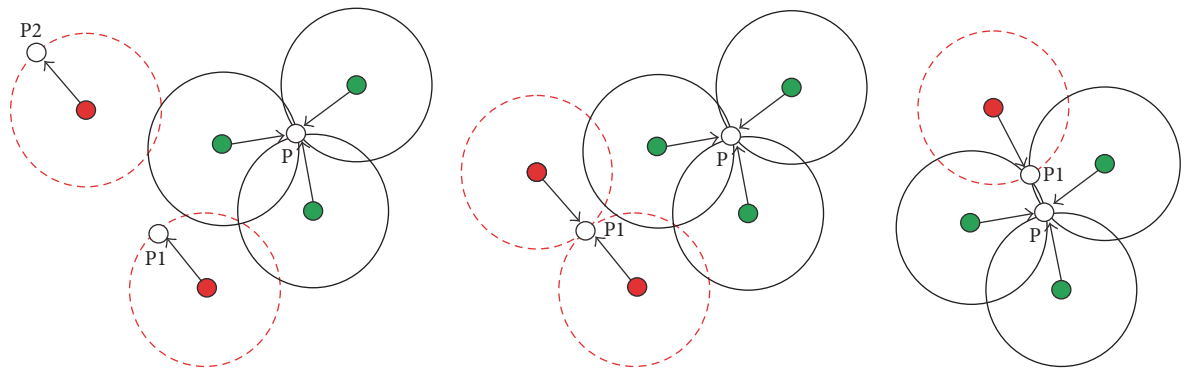
Figure 2: Effects of range change.

green nodes represent beacon nodes, and the white nodes represent common nodes.

In uncoordinated attack, different false location references are provided to mislead the unknown node to different false locations, for example, P1 and P2 in Figure 3(a). In collusion attack, all false location references mislead the common node to the same randomly chosen false location, say P1 in Figure 3(b). In pollution attack, all false location references misguide the unknown node, to a specially chosen false location P1, as in Figure 3(c), which still conforms to some normal location references. This attack succeeds even when normal location references are in the majority. In all the categories as shown in Figure 3, P is the original location.

3. Related Work

Whenever we talk about the secure localization [10] several related problems emerge like location privacy and location reporting. To mitigate the attacks on location identification or location calculation many researchers have proposed different schemes and approaches. They are classified into two types, node-centric and infrastructure-centric. Node-centric approaches deal with the calculation of information at node level. Based on their design goals, existing solutions can be further classified into three methods: (1) the prevention method, to prevent the adversaries from producing erroneous information, for example, HiRLOC [11], SeRLOC [12], ROPE [13], and SPINE [14]; (2) the detection method, to detect



(a) Uncoordinated (b) Collusion (c) Pollution

Figure 3: Location-reference attack variation.

and revoke the nodes producing erroneous information, for example, DRBTS [15], TSCD [16], and LAD [17]; and (3) the filtering method, to filter the received erroneous information in the location computation step such as ARMMSE [18] and i-Multihop [19]. On the other side, infrastructure-centric approaches emphasize the overall network structure for localization security, such as SLA [16] and SLS [20]. If a localization system is infrastructure-centric, the infrastructure will trust the estimation locations and no verification is needed, because the locations are computed by the infrastructure itself. However, if a localization system is node-centric, the nodes may be compromised and may intentionally report false locations. So the infrastructure may not simply trust the reported locations. Thus, when localization system is node-centric, location verification is a sound method for the infrastructure to check the validity of nodes' reported estimation. Different types of secure location verification methods [21] have been introduced such as Sector [22] and Distance Bounding Protocol [23].

Some of the recent research works in this direction have been identified. A very recent collaborative approach for secure localization has been shown in [9]. The proposed approach is based upon a trust model applied for under water wireless sensor networks. A cryptography based approach [24] is used for the secure localization using signature and encryption to provide confidentiality and integrity of the location information. It uses public key infrastructure along with Hash Message Authentication Code (HMAC) digest. Further, trilateration is used to calculate the coordinates of the unknown nodes. The proposed algorithm in [25] uses iterative gradient descent with selective pruning of inconsistent measurements to achieve high localization accuracy. The authors have also shown the accuracy of estimated location in mobile environment but have not emphasized the external nodes or elementary attacks. The proposed algorithm has not addressed the issue of false alarm. Different class of distance based localization algorithms have classified in [26]. The authors have also proposed a polynomial-time algorithm and two heuristic-based algorithms using a threshold value of the compromised nodes. A novel approach of secure localization has been observed in [27]. The authors have used Global Positioning System (GPS) systems and inertial guidance modules on special master node to provide the location accuracy. They have also used an efficient key distribution process in the algorithm. An encryption based secure localization algorithm is shown in [28]. The proposed algorithm, based on Paillier cryptosystem, provides a multilateral privacy preserving solution for secured least square estimation. A novel approach of secured localization using Connected Dominating Set (CDS) is discussed in [29]. Another secure localization technique is shown in [30]. The proposed method uses triangle inequality to detect the attack and then applies localization process based upon some reference points. Both processes use voting mechanism.

A novel approach of using game theory has been applied in [31]. The proposed algorithm combines two methods: Least Trimmed Square (LTS) algorithm is used in regression to identify and remove regression factors which are anomalous and Game Theoretic Aggregation (GTA) solves the problem of combining outputs from a number of predictors to generate a more accurate predictive model. To improve the performance of LTS, a single phase weight-based combination of factors is used by combining GTA with LTS, without any threshold specification. Another game based approach has been shown in [32]. The proposed approach uses trust evaluation and optimal payoff calculation to identify the strategy space of the nodes.

The use of decentralized dynamic key generation for secure localization has been researched in [33]. The proposed algorithm uses symmetric key encryption process with XOR operations and produces robustness with low overhead. A smart card based approach has been utilized in [34]. The proposed algorithm implements a secure and lightweight authentication scheme for heterogeneous wireless sensor networks using smart cards dynamic identities to prevent threats to users' privacy. Mutual trust in wireless sensor network has been discussed in [35]. The algorithm predistributes the random keys securely and uses identity based cryptography. Mutual trust is built up depending upon this identities and keys. A three-tier security framework is shown in [36]. The proposed framework uses two polynomial pools: the mobile

polynomial pool and the static polynomial pool. Authentication mechanism used between stationary access nodes and sensor nodes makes it more capable of withstanding to node replication attacks. The node capture attacks and flooding of packets in DV-hop localization are addressed in [37]. The proposed approach has used broadcast authentication and weight-based computation for secure localization purpose. A secure localization algorithm against wormhole attack has been discussed in [38]. The algorithm uses Round Trip Time (RTT) to collect information about the local subgraph. Ordinal Multidimensional Scaling (MDS) is used to adapt the topology changes. A verification method is also used here to minimize the false negatives. Another wormhole resistant localization solution has been observed in [39]. The algorithm uses different labels for pseudoneighbors and identifies the forbidden links. The algorithm is efficient in preventing the attack with the limitation that the nodes must have the identical radii. A number of approaches have been identified in the literature review. Almost all the existing works deal with the static network scenario. They also have a number of drawbacks such as extra hardware usage, more beacons, and control message transmission and predefined knowledge of the network topology. As per the need of mobility in the network environment, the security services in a mobile resource constrained environment are somehow critical to provide and therefore have received a less consideration in the previous works of the researchers. In this paper, we have provided a solution to the problem using an efficient certificate distribution and validation of distance estimation by the Base Station using a very less number of control messages. This will help for the WSNs to provide less overhead, better throughput, and better security from different types of attacks.

4. Proposed Algorithm

Our proposed algorithm considers only the anchor nodes, unknown nodes, and Base Station where anchor nodes and unknown nodes are deployed randomly. The anchors are having a variable range of transmission with an average transmission range R_{avg} given as

$$R_{\text{avg}} = \frac{\min \sum_{e \in E} \psi(|e|)}{m}, \tag{1}$$

where m is the number of anchor nodes in the network, e is an edge between two nodes, E is the set of the edges in the network, and $\psi(|e|)$ is the weighing function of a connection between an anchor node and an unknown node and interpreted as $\psi(|e|) \sim |e|^{\alpha}$, $2 \leq \alpha \leq 4$.

The algorithm starts with an initialization phase that deals with distribution of certificates by the BS. After the distribution of the certificates, distance estimation phase starts among the anchor nodes and the unknown nodes. Once the distances are estimated, the BS is able to localize the unknown nodes applying Minimum Mean Square Error (MMSE) method. The algorithm is summarized in Algorithm 1.

As we have used the speed of light, c, to estimate the distance, the process shown above will prevent the generation of high speed link required to execute wormhole attack because there cannot be any high speed link in which the transmission speed will be more than that of the light. The utilization of mutual authentication with certificates provided by the BS will help to avoid or prevent any kind of authentication attack such as Sybil attack and impersonation attack executed by the outsider nodes. The encryption method will help to securely transmit the estimated distance to the BS. The $t_{\text{retransmit}}$ value will help to detect the jamming attack so that further the avoidance and detection process can applied following the methods as shown in [40]. But it can be a fact that the insider nodes are compromised and can generate distance reduction or enlargement attacks. To prevent these attacks, we have to follow the further process.

Let us assume that the deviation of the true position of the unknown node due to measurement error and/or malicious distance estimates is δ which is tolerable for the system. We know that the unknown node (x_{u_i}, y_{u_i}) must be in the intersection region of the anchor nodes' bound circles in the range. Therefore, in Algorithm 2 we can validate the distance estimation provided by the anchor nodes.

5. Network Model and Assumptions

The network model is considered to be self-organizing having no central control of deploying the sensor nodes in the network. For the ease of presentation, the wireless sensor network model $\mathcal{N}$ is considered to be in 2D and represented by a graph $G(V, E)$ which consists of V, a set of vertices, and E a set of edges. The size of the network can be given as

$$|\mathcal{N}| = |A| + |U|, \tag{2}$$

where $|A|$ is the size of anchor node set A, $|U|$ is the size of the unknown node set U, and $A, U \subseteq V$.

In the proposed algorithm, we have divided the network nodes in two categories of nodes. First, the anchor nodes, $a_j \in A$, which are privileged in their storage capacity and computational capacity with additional energy resources. Secondly, the unknown nodes $u_i \in U$, which are not privileged like the anchor nodes and are able to perform minimum computational tasks. Both types of nodes are randomly deployed in the network environment. The location estimation of an unknown node is calculated by using the location information of the anchor nodes in a WSN. Therefore, the integrity of location messages as well as the reliability of message origin is very important during the localization process. Confidentiality of estimated location is also required in some applications, to protect the privacy of the corresponding sensors. In this paper, an appropriate cryptographic scheme is presented to provide the security services. The assumptions for our proposed approach have been listed below.

(i) The unknown nodes and anchor nodes are mobile.

(ii) Base Station (BS) is assumed to be trusted and is considered to be key distributor and certificate authority.

(iii) Anchor nodes and unknown nodes are deployed with their private keys.

Input. anchor node set A, unknown node set U
Step 1. BS creates identities ID_{a_j} for all anchor nodes and identities ID_{u_i} for all unknown nodes
Step 2. BS provides certificates: Cert_{a_j}, Cert_{u_i}
Step 3. $\forall a_j \in A$ do
a_j sends u_i random nonce $\varkappa$, Cert_{a_j}; for $i = 1, 2, \ldots, n$ and $j = i = 1, 2, \ldots, m$
a_j waits for a threshold time $t_{\mathrm{retransmit}}$ to retransmit the message
Step 4. $\forall u_i$ under R_{avg} for *any* $a_j \in A$
u_i sends a_j: $[\varkappa, \mathrm{time}_{\mathrm{proc}_u}]_{K_{a_j+}}$, Cert_{u_i}
Step 5. Calculate $\mathrm{time}_{\mathrm{prop}}$
Step 6. $d_{u_i}^{a_j} = c \times \mathrm{time}_{\mathrm{prop}}$
Step 7. a_j sends $d_{u_i}^{a_j}$ to the Base Station (BS)
Step 8. end loop
Step 9. Apply MMSE

Algorithm 1: Distance estimation by anchor nodes.

Input. Set of anchor nodes A with locations (x_{a_j}, y_{a_j}), location estimate of an unknown node (x_{u_i}, y_{u_i}), error parameter δ
Step 1. $\forall a_j \in A$, $j = 1, 2, \ldots, m$
If $(\mathrm{true}_{d_{u_i}^{a_j}} - \delta)^2 \leq (x_{u_i} - x_{a_j})^2 + (y_{u_i} - y_{a_j})^2 \leq (\mathrm{true}_{d_{u_i}^{a_j}} + \delta)^2$
then exit
else go to Step 2
Step 2. calculate the algebraic centre x^* of intersection region $\mathscr{R}$
Step 3. Initialize $r^* = 0$ //radius of the intersection region $\mathscr{R}$ as
Step 4. $\forall v$ inside the region $\mathscr{R}$ do
if $\|v - r^*\| > r^*$
then $r^* \leftarrow \|v - r^*\|$
end if
Step 5. $\forall a_j \in A$, $j = 1, 2, \ldots, m$ do
$\overline{\overline{\mathrm{true}_{d_{u_i}^{a_j}}}} = \dfrac{\mathrm{true}_{d_{u_i}^{a_j}}}{1 + \varepsilon_{\max}}$
Step 6. if $\overline{\overline{\mathrm{true}_{d_{u_i}^{a_j}}}} > \|x^* - a_j\| + r^*$ then
Anchor node a_j is malicious
else
a_j is not malicious
Step 7. end if

Algorithm 2: Validation of distance estimation and detection of malicious anchors by BS.

(iv) Base Station (BS) shares the public key only to the legitimate unknown nodes and anchor nodes predefined.

Initialization Phase. Base Station (BS) provides the identity for all anchor nodes and unknown nodes as ID_{a_j} and ID_{u_i} where a_j is an anchor node and u_i is an unknown node. BS also provides certificates for each anchor node and unknown node as Cert_{a_j} and Cert_{u_i}.

$$\mathrm{BS} \longrightarrow \mathrm{Cert}_{a_j} = \left[\mathrm{ID}_{a_j}, K_{a_j+}, t, e_t\right] \mathrm{BS}_{K-}, \tag{3}$$

where ID_{a_j} is the identity of an anchor node a_j, K_{a_j+} is the public key of that anchor node, t is the timestamp when the certificate was created, and e_t is the expiry time of the certificate. This total certificate is digitally signed by BS_{K-} which is the private key of the Base Station. All anchor nodes must make them update themselves by having a fresh certificate as required. For an legitimate unknown node u_i, we can rewrite the above format in the following way:

$$\mathrm{BS} \longrightarrow \mathrm{Cert}_{u_i} = \left[\mathrm{ID}_{u_i}, K_{u_i+}, t, e_t\right] \mathrm{BS}_{K-}, \tag{4}$$

where ID_{u_i} is the identity of an unknown node u_i, K_{u_i+} is the public key of that unknown node, and e_t is the expiry time of the certificate.

Distance Estimation Phase. The anchor node a_j sends a random nonce $\varkappa$, along with the certificate Cert_{a_j} to all the one-hop neighborhood unknown nodes u_i in the range R_{avg}

and starts the timer on. When the unknown nodes receive the message, verify the certificate using the public key BS_{K+} given by BS. As, only legitimate anchor nodes are having the certificate to provide, by verifying the certificates, the authentication of the anchor nodes can be proved. Then, the unknown nodes u_i response back to the anchor node a_j with the same nonce $\varkappa$, time duration between of receiving the last bit of message sent by anchor node and transmitting the first bit of message to the anchor node, given as $\text{time}_{\text{proc}_u}$ encrypted with anchor node's public key K_{a_j+} along with its own certificate.

$$\begin{aligned} a_j &\longrightarrow u_i : \varkappa, \text{Cert}_{u_i}, \\ u_i &\longrightarrow a_j : \left[\varkappa, \text{time}_{\text{proc}_u}\right]_{K_{a_j+}}, \text{Cert}_{u_i}. \end{aligned} \tag{5}$$

When a_j sends message to u_i, it waits for a bounded time value $t_{\text{retransmit}}$ to retransmit the message if no response starts arriving to the anchor in that bounded time. This value is precomputed at the starting of the network deployment assuming all the favourable conditions of the network environment with a noise effect of Δt and given as

$$t_{\text{retransmit}} = \text{time}_{\text{normal}} + \Delta t, \tag{6}$$

where $\text{time}_{\text{normal}}$ is the normal time duration of getting a response back from the unknown node.

When the anchor node receives the response back from the unknown nodes, it decrypts the message using its own private key K_{a_j-}, verifies the certificate of the unknown nodes, stops the timer, and calculates the signal propagation time as

$$\text{time}_{\text{prop}} = \frac{\left(\text{time}_j - \text{time}_{\text{proc}_u} - \text{time}_{\text{proc}_a}\right)}{2}, \tag{7}$$

where $\text{time}_{\text{prop}}$ is the signal propagation time, time_j is the timer interval at the anchor side, and $\text{time}_{\text{proc}_a}$ is the time duration between receiving the first bit of the response and last bit of the response. The interaction between unknown node and anchor node is shown in Figure 4.

Once the propagation time is calculated, the estimated distance between anchor node a_j and unknown node u_i is calculated as

$$d_{u_i}^{a_j} = c \times \text{time}_{\text{prop}}, \quad \text{where } c \text{ is the speed of light.} \tag{8}$$

Once the anchor node calculates this estimated distance, it is then forwarded to the BS encrypted with the public key of BS and along with the anchor node's certificate.

$$a_j \longrightarrow \text{BS} : \left[d_{u_i}^{a_j}\right]_{\text{BS}_{K+}}, \text{Cert}_{a_j}. \tag{9}$$

After receiving the message from the anchor nodes, BS decrypts the message with is private key and gets the estimated distances. Finally, it uses Minimum Mean Square Error (MMSE) [41] to estimate the location of an unknown node (x_{u_i}, y_{u_i}). One thing needs to remember is that we need at least three noncollinear anchor nodes to apply MMSE. Another important attribute of our proposed algorithm deals with the mobility of the nodes. We consider that the nodes (whether the anchor or the unknown) are mobile. The relative mobility between an unknown node u_i and anchor node a_j at a given time t is given by

$$\text{RM}_t^{a,u} = d_{a,u_t} - d_{a,u_{t-1}} \tag{10}$$

$\text{RM}_t^{a,u}$ is positive if node u_i is moving away from a_j and negative if u_i is coming closer to a_j.

Though the mobility is incorporated in the algorithm, nodes (both the anchor nodes and the unknown nodes) are assumed to be pseudostatic; that is, they are static for a very short time interval for the localization process and this does not incorporate any significant error in the estimation.

Handling Distance Estimation Error. Distance estimations in a wireless environment are very common to have error due to the noise or delay in the medium. Assume that the estimation error is $\epsilon \in [-\epsilon_{\max}, \epsilon_{\max}]$, where $\epsilon_{\max}$ is a system parameter and given as $0 \le \epsilon_{\max} \le 1$. Therefore, the estimated distance can be given as

$$d_{u_i}^{a_j} \in \left[\text{true}_{d_{u_i}^{a_j}} \times \left(1 - \epsilon_{\max}\right), \text{true}_{d_{u_i}^{a_j}} \times \left(1 + \epsilon_{\max}\right)\right], \tag{11}$$

where $\text{true}_{d_{u_i}^{a_j}}$ is the true distance between a_j and u_i and can be calculated by applying Euclidean method.

Further, the presence of compromised insider anchor nodes can create an error factor θ. Following this, the estimated distance between a_j and u_i in presence of malicious anchor node can be given as

$$d_{u_i}^{a_j} = \text{true}_{d_{u_i}^{a_j}} \times \left(1 + \epsilon_{\max}\right) \times \left(1 + \theta\right), \quad \text{for } \theta > 0. \tag{12}$$

As we know that $\epsilon \in [-\epsilon_{\max}, \epsilon_{\max}]$, the value of ϵ can create both the positive estimation error and negative estimation error. Positive estimation error will create multiple intersection points of the convex region of the anchor nodes' ranges leading to the distance enlargement attacks. On the other hand, negative estimation error creates an empty intersection region assuming that the location of the unknown node is in the intersection of bounds of anchors leading to the distance reduction attack. This concept is shown in Figure 5. The black solid circles are anchor nodes and green circle is the original estimated location. If the anchor nodes are compromised and provide reduced distance estimations, the intersection will be empty and if the malicious anchor nodes provide enlarged distance estimations, the position of the unknown node deviates from the original position shown as light blue circle.

Distance reduction is not a severe in WSN localization. If we find the empty intersection region $\mathscr{R}$, the distance estimates can be increased with a factor of $1/(1-\epsilon_{\max})$ to get a nonempty intersection region $\mathscr{R}'$, where the unknown node must exist.

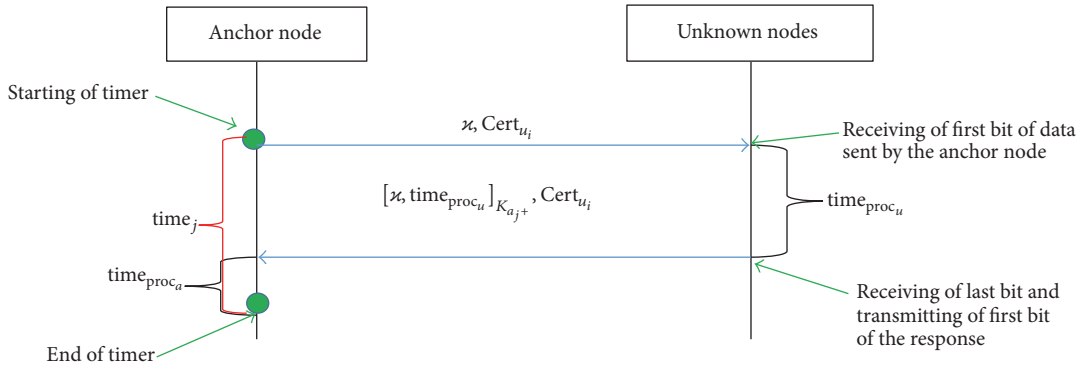


Figure 4: Propagation time estimation process.

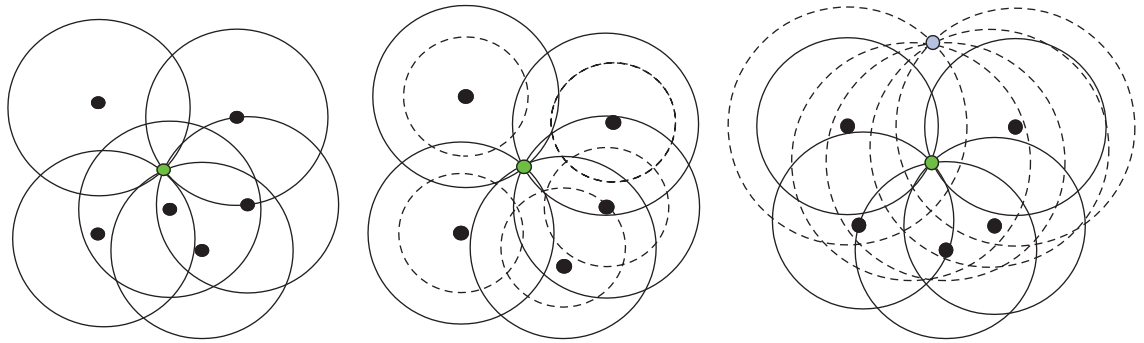


Figure 5: (a) Truthful estimation. (b) Distance reduction. (c) Distance enlargement.

To prevent distance enlargement situation, the BS need to follow the process summarized in Algorithm 2. The tolerable error parameter δ can be derived from the following equation as

$$\delta = w_1\epsilon + w_2\theta, \tag{13}$$

where ϵ is the system measurement error due to noise and θ is the error included by malicious anchor nodes. We assume that the unknown nodes are error free and do not provide any false distance estimation. w_1, w_2 are used as weighing values for the errors depending upon the network conditions. This δ will provide an upper bound and lower bound of the estimated distance in presence of error given as

$$\begin{aligned}\left(\text{true}_{d_{u_i}^{a_j}} - \delta\right)^2 &\le \left(x_{u_i} - x_{a_j}\right)^2 + \left(y_{u_i} - y_{a_j}\right)^2 \\ &\le \left(\text{true}_{d_{u_i}^{a_j}} + \delta\right)^2,\end{aligned} \tag{14}$$

$$\text{true}_{d_{u_i}^{a_j}} = \sqrt{\left(x_{u_i} - x_{a_j}\right)^2 + \left(y_{u_i} - y_{a_j}\right)^2}.$$

The algebraic centre x^* in Algorithm 2 can be calculated using barrier method on the unconstrained optimization problem given as

$$\begin{aligned}\min \quad & (x, \delta) - \lambda \cdot \delta \\ & - \sum_{j=1}^{m} \log\left[\left(\overline{\text{true}_{d_{u_i}^{a_j}}} \cdot (1 - \delta)\right)^2 - \left\|x - a_j\right\|^2\right] \\ & - \log(\delta),\end{aligned} \tag{15}$$

where λ is the Lagrangian multiplier and $\overline{\text{true}_{d_{u_i}^{a_j}}}$ is given by $\overline{\text{true}_{d_{u_i}^{a_j}}} = \text{true}_{d_{u_i}^{a_j}}/(1 - \varepsilon_{\max})$, that is, the increased distance estimation in case of negative estimation error.

The radius of the intersection region $\mathscr{R}$ is initialized with 0 with an assumption that the unknown node is positioned at the intersection point itself and no convex region has been generated by the intersection. Moreover, the radius of the intersection region can be updated by verifying the distance between any point ν inside the region and the algebraic centre x^*. Finally, we can detect the malicious insider anchor nodes depending upon the increased estimated distance.

So the attacks, those are identified in localization process as shown in Table 1, are addressed in the proposed model. The

Table 2: Prevention of attacks by the proposed model.

Attacks	Attack behaviour	Prevention by our proposed model
Stealing	Signal eavesdropping and tampering	Our proposed model uses encryption to prevent such attacks
Jamming	Sending jamming signal in the working frequency range	Detection is addressed in the proposed algorithm
Collision	Repetition of messages	Not applicable in the proposed model, as the maximum calculation is done by BS and anchor node with minimum message controls
Exhaustion	Sending of unnecessary message	No scope to provide unnecessary message as transmission range is limited to and the distance estimation process is secured
Unfairness	Explicitly taking the control of the channel	Not possible due to the minimum size of the packets
DoS Attacks	Exhaustion of energy of the unknown nodes	Can be monitored directly by the Base Station
Selective forwarding	Selectively forward packets	Using the approach of one-hop neighborhood forwarding is not necessary
Sybil	Possessing multiple identities	Mutual authentication is used
Sinkhole	Maliciously tamper with routing	Mutual authentication is used with the certificates
Wormhole	Shortening the distance to make a fast routing path	The distance estimation is done based upon the light speed which is the maximum speed of transmission can be and therefore no faster route can be created between an anchor and an unknown node
Flooding	Establishing false connections	Broadcasting is limited by the anchor nodes within a limited range of R_{avg}
Tampering	Tampering localization beacons	Both encryption and mutual authentication are used
Insider attack	Compromised anchor nodes may provide false information	Both the distance reduction and distance enlargement attack have been addressed
Range change attack	Changing the range or Angle of Arrival (AoA)	Our proposed model does not incorporate the mechanism of AoA as it works on time interval to calculate the distance and therefore can easily avoid such attack
False beacon location attack	Compromising a beacon and then he can make the beacon broadcast false location	Authentication, limited range, and validation of distance estimation in the proposed approach will help to avoid such attack
False reported location attack	Malicious node reports false	Verification is done at the BS, so there is less chance to report falsified verification

Table 3: Simulation parameters.

Simulation area	500 m × 500 m
Number of unknown nodes	500
Communication range	120 m
Node deployment	Random
Mobility model	Random Way Point model

summarization of countermeasures by our proposed model has been shown in Table 2.

6. Results and Discussion

In this section, we have evaluated the proposed algorithm based on the parameters as shown in Table 3.

We have compared the simulated results with the three recent algorithms: (1) Collaborative Secure Localization algorithm based on Trust model (CSLT) proposed by Han et al. [9], (2) Multilateral Privacy Algorithm (MPA) for secured localization proposed by Shu et al. [28], and (3) Authenticated Weight-based Secured (AWS) DV-hop proposed by Liu et al. [37]. The performances of the algorithms are measured on the following three parameters: localization efficiency, localization accuracy, and malicious detection ratio.

The attacks described in Table 2 are also simulated to show the efficiency of the proposed algorithm. The localization ratio is defined as the percentage of successful location estimation of unknown nodes. The result in Figure 6(a) shows that, with the increasing malicious nodes' percentage, every algorithm in our comparison faces a significant decrease in successful localization of unknown nodes. However, the proposed algorithm still performs better as compared to others. Figure 6(b) shows that the proposed algorithm outperforms the other algorithms in the successful localization of unknown nodes with the increasing percentage of anchor nodes. Localization accuracy is a valuable metric for evaluating the efficiency of localization algorithms.

In the proposed work, the localization accuracy is defined by the relative error between the actual location and the calculated node position. In our simulation, we have varied the ratio of malicious nodes from 5% to 30% with increments

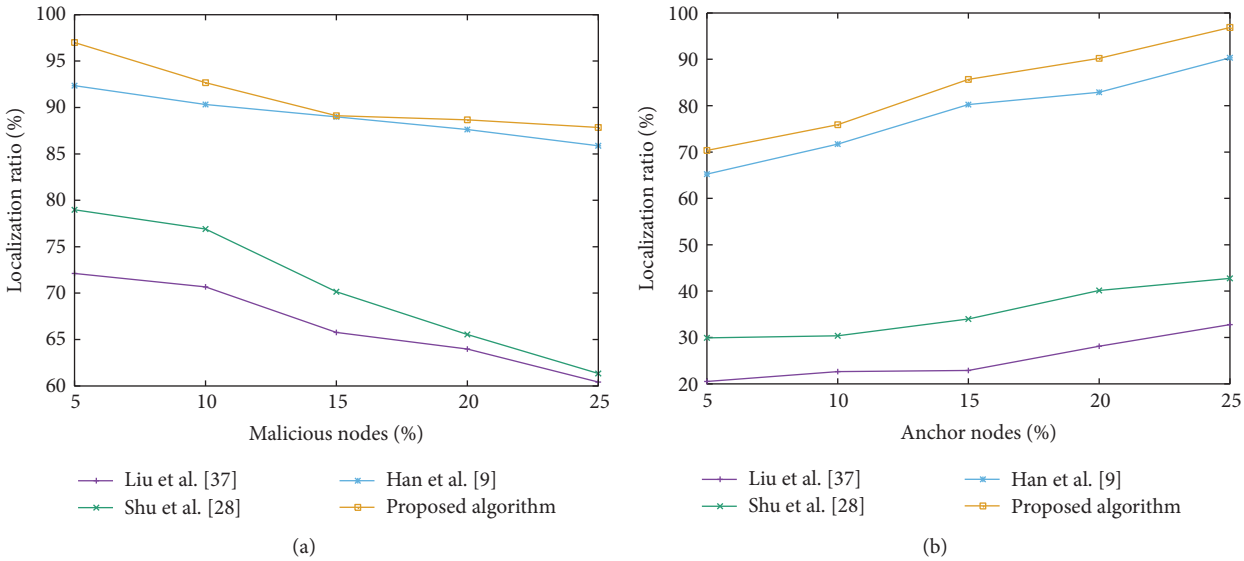


Figure 6: Comparison of localization ratio: (a) impact of malicious nodes and (b) impact of anchor nodes.

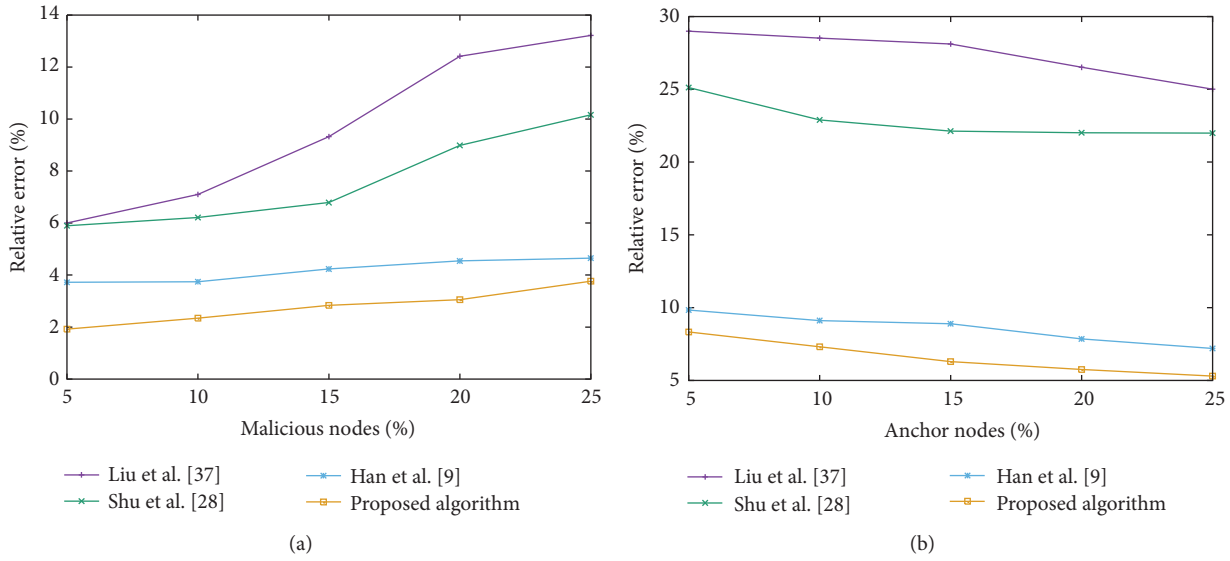


Figure 7: Comparison of localization accuracy: (a) impact of malicious nodes and (b) impact of anchor nodes.

of 5%. Simulation result, shown in Figure 7(a), shows that the relative error percentage of location estimation increases with the increasing number of malicious nodes. However, the proposed algorithm proves its efficiency in location estimation accuracy. Similarly, location accuracy is also tested by varying the anchor nodes' percentage. Result shown in Figure 7(b) signifies to the fact that the proposed algorithm significantly reduces the relative error percentage with the increasing number of anchor nodes. It is also seen in the result that the other algorithms also decrease the relative error with the increasing number anchor nodes, but the percentage of relative error is less in our proposed algorithm.

Simulation time is defined as the time taken for the algorithms to detect a particular malicious attack. The result in Figure 8 shows that the proposed algorithm is efficient in detecting 90% of the malicious attack with less time as compared to the other algorithms in comparison.

7. Conclusion

Security in localization has always been a vital part of localization algorithms. Though there are a number of algorithms which are introduced with security aspects, but the algorithm designers have somehow overlooked the complexity issue of the algorithms in the resource constrained WSNs. In this paper, we have addressed this problem and provided a solution with our proposed algorithm. The proposed algorithm not only prevents a number of outsider attacks but

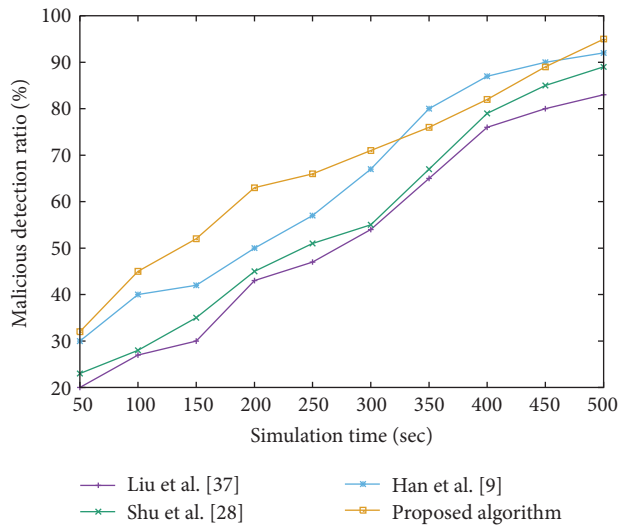


Figure 8: Comparison of malicious detection ratio.

also provides a check on the insider nodes. Moreover, the algorithm provides low overhead and major functionality is based on Base Station. The simulation results also prove the efficiency of the proposed algorithm in terms of localization efficiency, localization accuracy, and malicious detection ratio. The most important feature of our algorithm is that it supports mobility of the nodes and therefore it is suitable for dynamic network environments.

Competing Interests

The authors declare that they have no competing interests.

References

[1] D. Niculescu and B. Nath, "Localized positioning in ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 247–259, 2003.

[2] S. Meguerdichian, S. Slijepcevic, V. Karayan, and M. Potkonjak, "Localized algorithms in wireless ad-hoc networks: location discovery and sensor exposure," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp. 106–116, 2001.

[3] D. D. Perkins, R. Tumati, H. Wu, and I. Ajbar, "Localization in wireless ad hoc networks," in *Resource Management in Wireless Networking*, pp. 507–542, Kluwer Academic Publishers, Boston, Mass, USA, 2005.

[4] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Vehicular ad hoc networks: a new challenge for localization-based systems," *Computer Communications*, vol. 31, no. 12, pp. 2838–2849, 2008.

[5] K. K. Chintalapudi, "On the feasibility of Ad-Hoc localization systems," Tech. Rep. 117, Computer Science Department, University of Southern California, Los Angeles, Calif, USA, 2003.

[6] G. Kumar and M. K. Rai, "An energy efficient and optimized load balanced localization method using CDS with one-hop neighbourhood and genetic algorithm in WSNs," *Journal of Network and Computer Applications*, vol. 78, pp. 73–82, 2017.

[7] X.-M. Cao, B. Yu, G.-H. Chen, and F.-Y. Ren, "Security analysis on node localization systems of wireless sensor networks," *Journal of Software*, vol. 19, no. 4, pp. 879–887, 2008.

[8] J. Jiang, G. Han, C. Zhu, Y. Dong, and N. Zhang, "Secure localization in wireless sensor networks: a survey (Invited Paper)," *Journal of Communication*, vol. 6, p. 123, 2011.

[9] G. Han, L. Liu, J. Jiang, L. Shu, and J. J. P. C. Rodrigues, "A collaborative secure localization algorithm based on trust model in underwater wireless sensor networks," *Sensors*, vol. 16, no. 2, article 229, 2016.

[10] A. Srinivasan and J. Wu, "A survey on secure localization in wireless sensor networks," in *Encyclopedia of Wireless and Mobile Communications*, p. 126, CRC Press, 2007.

[11] L. Lazos and R. Poovendran, "HiRLoc: high-resolution robust localization for wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 233–246, 2006.

[12] L. Lazos and R. Poovendran, "SeRLoc: secure range-independent localization for wireless sensor networks," in *Proceedings of the 3rd ACM Workshop on Wireless Security (WiSe '04)*, pp. 21–30, Philadelphia, Pa, USA, October 2004.

[13] L. Lazos, R. Poovendran, and S. Čapkun, "ROPE: robust position estimation in wireless sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 324–331, IEEE, April 2005.

[14] S. Capkun and J. Hubaux, "Secure positioning in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 221–232, 2006.

[15] A. Srinivasan, J. Teitelbaum, and J. Wu, "DRBTS: distributed reputation-based beacon trust system," in *Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC '06)*, pp. 277–283, IEEE, Indianapolis, Ind, USA, October 2006.

[16] F. Anjum, S. Pandey, and P. Agrawal, "Secure localization in sensor networks using transmission range variation," in *Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS '05)*, pp. 195–203, Washington, DC, USA, November 2005.

[17] W. Du, L. Fang, and P. Ning, "LAD: localization anomaly detection for wireless sensor networks," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS '05)*, p. 41, IEEE, April 2005.

[18] D. Liu, P. Ning, and W. K. Du, "Attack-resistant location estimation in sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 99–106, IEEE, April 2005.

[19] C. Wang and L. Xiao, "Sensor localization in concave environments," *ACM Transactions on Sensor Networks*, vol. 4, no. 1, article 3, 2008.

[20] Y. Zhang, W. Liu, Y. Fang, and D. Wu, "Secure localization and authentication in ultra-wideband sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4 I, pp. 829–835, 2006.

[21] N. Sastry, U. Shankar, and D. Wagner, "Secure verification of location claims," in *Proceedings of the 2nd ACM Workshop on Wireless Security (WiSE '03)*, p. 110, San Diego, Calif, USA, 2003.

[22] S. Čapkun, L. Buttyn, and J.-P. Hubaux, "SECTOR: secure tracking of node encounters in multi-hop wireless networks," in *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 21–32, 2003.

[23] K. B. Rasmussen and S. Čapkun, "Location privacy of distance bounding protocols," in *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS '08)*, pp. 149–160, ACM, Alexandria, VA, USA, October 2008.

[24] T. Zhang, J. He, X. Li, and Q. Wei, "A signcryption-based secure localization scheme in wireless sensor networks," *Physics Procedia*, vol. 33, pp. 258–264, 2012.

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

[26] M. Jadliwala, S. Zhong, S. J. Upadhyaya, C. Qiao, and J.-P. Hubaux, "Secure distance-based localization in the presence of cheating beacon nodes," *IEEE Transactions on Mobile Computing*, vol. 9, no. 6, pp. 810–823, 2010.

[27] Q. Mi, J. A. Stankovic, and R. Stoleru, "Practical and secure localization and key distribution for wireless sensor networks," *Ad Hoc Networks*, vol. 10, no. 6, pp. 946–961, 2012.

[28] T. Shu, Y. Chen, and J. Yang, "Protecting multi-lateral localization privacy in pervasive environments," *IEEE/ACM Transactions on Networking*, vol. 23, no. 5, pp. 1688–1701, 2015.

[29] A. Srinivasan, "SecLoc—secure localization in WSNs using CDS," *Security and Communication Networks*, vol. 4, no. 7, pp. 763–770, 2011.

[30] W. T. Zhu, Y. Xiang, J. Zhou, R. H. Deng, and F. Bao, "Secure localization with attack detection in wireless sensor networks," *International Journal of Information Security*, vol. 10, no. 3, pp. 155–171, 2011.

[31] S. Jha, S. Tripakis, S. A. Seshia, and K. Chatterjee, "Game theoretic secure localization in wireless sensor networks," in *Proceedings of the International Conference on the Internet of Things (IOT '14)*, pp. 85–90, IEEE, Cambridge, Mass, USA, October 2014.

[32] T. Bao, J. Wan, K. Yi, and Q. Zhang, "A game-based secure localization algorithm for mobile wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 642107, 8 pages, 2015.

[33] Z. Merhi, A. Haj-Ali, S. Abdul-Nabi, and M. Bayoumi, "Secure localization for wireless sensor networks using decentralized dynamic key generation," in *Proceedings of the 8th International Wireless Communications and Mobile Computing Conference (IWCMC '12)*, pp. 543–548, 2012.

[34] C.-C. Chang, W.-Y. Hsueh, and T.-F. Cheng, "A dynamic user authentication and key agreement scheme for heterogeneous wireless sensor networks," *Wireless Personal Communications*, vol. 89, no. 2, pp. 447–465, 2016.

[35] C.-H. Lin, Y.-H. Huang, A. D. Yein, W.-S. Hsieh, C.-N. Lee, and P.-C. Kuo, "Mutual trust method for forwarding information in wireless sensor networks using random secret pre-distribution," *Advances in Mechanical Engineering*, vol. 8, no. 4, pp. 1–9, 2016.

[36] A. Rasheed and R. N. Mahapatra, "The three-tier security scheme in wireless sensor networks with mobile sinks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 5, pp. 958–965, 2012.

[37] X. Liu, R. Yang, and Q. Cui, "An efficient secure DV-Hop localization for wireless sensor network," *International Journal of Security and Its Applications*, vol. 9, no. 7, pp. 275–284, 2015.

[38] S. Mukherjee, M. Chattopadhyay, S. Chattopadhyay, and P. Kar, "Wormhole detection based on ordinal MDS using RTT in wireless sensor network," *Journal of Computer Networks and Communications*, vol. 2016, Article ID 3405264, 15 pages, 2016.

[39] H. Chen, W. Lou, Z. Wang, J. Wu, Z. Wang, and A. Xi, "Securing DV-Hop localization against wormhole attacks in wireless sensor networks," *Pervasive and Mobile Computing*, vol. 16, pp. 22–35, 2015.

[40] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *IEEE Network*, vol. 20, no. 3, pp. 41–47, 2006.

[41] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pp. 166–179, July 2001.

Hindawi
Mobile Information Systems
Volume 2017, Article ID 7251395, 7 pages
https://doi.org/10.1155/2017/7251395

Research Article

Location Privacy Protection Based on Improved K-Value Method in Augmented Reality on Mobile Devices

Chunyong Yin, Jinwen Xi, and Ruxia Sun

*School of Computer and Software, Jiangsu Engineering Center of Network Monitoring,
Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology,
Jiangsu Key Laboratory of Meteorological Observation and Information Processing,
Nanjing University of Information Science & Technology, Nanjing 210044, China*

Correspondence should be addressed to Ruxia Sun; src@nuist.edu.cn

Received 29 October 2016; Accepted 27 December 2016; Published 13 February 2017

Academic Editor: Jaegeol Yim

With the development of Augmented Reality technology, the application of location based service (LBS) is more and more popular, which provides enormous convenience to people's life. User location information could be obtained at anytime and anywhere. So user location privacy security suffers huge threats. Therefore, it is crucial to pay attention to location privacy protection in LBS. Based on the architecture of the trusted third party (TTP), we analyzed the advantages and shortages of existing location privacy protection methods in LBS on mobile terminal. Then we proposed the improved K-value location privacy protection method according to privacy level, which combines k-anonymity method with pseudonym method. Through the simulation experiment, the results show that this improved method can anonymize all service requests effectively. In addition to the experiment of execution time, it demonstrated that our proposed method can realize the location privacy protection more efficiently.

1. Introduction

Augmented Reality technology, called AR, is a kind of technology that real time calculates the position and angle of the camera image and allows adding the corresponding image, video, 3D model generated by a computer to the reality [1]. The concept of AR was proposed in 1990 by Thomas Caudell, an employee of Boeing [2]. After many years of technological development, AR has evolved through different stages and now it is becoming one of the commonly used technologies. Today one of the commonly accepted definitions of AR is given by Ronald Azuma [3], which says that Augmented Reality contains three aspects:

(1) Combination of virtualness and reality

(2) Real-time interactivity

(3) Registration in 3D

With the development of wireless sensor technology and advanced devices, it is possible to get the accurate personal location information of the mobile terminal user anytime and anywhere; therefore, location based service (LBS) is a new class of applications. Location based service is one of the common services provided by AR, which accesses to the position information related to the user through mobile wireless network or external positioning mode [4]. With the information, it adds the value of services. LBS normally consists of location system, mobile devices, network, and service provider (LBS server). In this service, users send the local position information to LBS server and get the corresponding query results [5]. For example, the user uses the mobile phone to send the information to the server, then the location system acquires the query, and finally the server returns the feedback to the user through network.

There are many categories of services that LBS can provide, including Emergency Services, Communities and Entertainment, Information and Navigation, Tracking and Monitoring, and Mobile Electronic Commerce. In 2003, CSTB (Computer Science and Telecommunications Board) in the "IT Roadmap to a Geospatial Future" pointed that LBS would be a very important part of future computing

environment, with the gradual maturity of technology, and it would be infiltrated into all aspects of the future life. The ABI research of market research firm forecasted that the global number of people enjoying location based services from 1.2 million in 2006 would increase to 31.5 million in 2011. And now, the number is much more than that.

The head mounted display is used as a fusion display device in early Augmented Reality system, which limits the scope of the user's activities to a certain extent and is not conducive to the outdoor environment. With the rapid development of mobile devices and network technology, the application of Augmented Reality technology in mobile terminals has been involved in many fields, such as games, social networks, e-commerce, and personal health care. So it is very important to classify these services that are achieved from those fields. KNN algorithm, SVM algorithm [6, 7], Hoeffding-ID data-stream [8], and so on are the common classification algorithms. At the same time, feature selection is also necessary in the process of classification. Some effective feature selection algorithms are introduced in [9].

The application of AR is becoming more and more extensive with the development of AR and LBS technology. In other words, a technology boom takes place in the case of AR and LBS is one of the most widely used services of AR. For example, Pokémon GO is the popular game based on AR, which springs up around the world. However, if users do not take the appropriate security measures and develop these technologies unlimitedly, widely known serious privacy threats will be presented to them. The important threats are the leak of service content and location privacy. Service content threat is the potential exposure of service users. For example, a user searches the Internet regularly; he does not want to be identified as the subscriber of some LBS. User's location is disclosed in the service request, which results in the leak of location privacy. Some sensitive information may be revealed such as health conditions and lifestyles. The leak of location privacy restricts the use of LBS, which has also become the bottleneck of the development of LBS and AR technology [10].

2. Related Work

User location privacy [11] is a kind of special information privacy, and it still belongs to the category of information privacy. Information privacy refers to sharing information with others in a certain period of time, in a place or in some way defined by individuals or institutions, and location privacy refers to preventing the attacker from accessing to the user's location information in some way as far as possible. In LBS, sensitive attribute data can be the time information and spatial information related to the users and the content of the service request contains many respects, such as health care information and property information. The attackers can use this information to infer the user's travel patterns, hobbies and interests, and other personal privacy information. Location privacy threat refers to, under unauthorized circumstance, the fact that attacker tracks the original position information through location device and technology and infers the privacy information related to user location through reasoning [12].

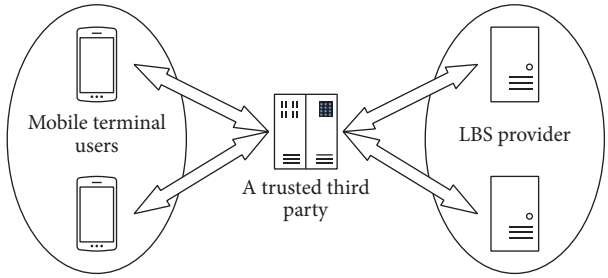


Figure 1: The trusted third party model.

Location privacy protection method mainly refers to the fact that the user provides false user location privacy information or anonymous user's identity information and location information to the server in the process of location service. The model of location privacy protection is divided into 2 categories, which are trusted third party (TTP, shown in Figure 1) and free trusted third party (FTTP) [13]. This paper only discusses the previous class method.

It is easy to understand the location privacy protection based on TTP model; as a result, it is very common. Generally speaking, a reasonable discount is proposed among the efficiency, accuracy, and privacy. In order to solve the problem of location privacy leakage, many researchers try to balance the service quality and privacy protection, which means the best service with least location privacy exposure.

Today, we have already proposed a lot of privacy protection methods, like pseudolocation method, pseudonym method, k-anonymity method, and other methods based on it, such as personalized k-anonymity.

2.1. Pseudolocation Method. Pseudolocation method is another location privacy protection method, which is used to protect the user's identity. This technology generalizes the user's true location and uses the location space region coordinates to represent the user's true location information [14]. There are two situations for this method to realize the location privacy protection. The first one is that the user forms the pseudolocation by himself, after putting forward the service request, and sends it with his real location to the LBS provider. Therefore, the attacker cannot discriminate the pseudo and real location, which protects the user's location privacy. The other situation is that when putting forward the service request, the user only sends one specified pseudolocation. When the server receives the location, it will increase the resent adjacent inquiry and send the results to the client. So the user can find the requisite answer from the results. In this way, the location privacy can still be protected because the real location information of the user is not acquired by the attacker. But the defect of this method is still obvious. In this method, the space of user acts is restricted. And the level of location privacy protection is not fixed, which is proportional to the distance between the pseudo- and real locations. In other words, the level of privacy protection will

be low if the pseudolocation is close to the real location of the user and vice versa.

2.2. Pseudonym Method. Pseudonym method [15] can realize the protection of user identity. That is to say, the user sends a service request through a false identity instead of the true identity and confuses the relationship between the position information and user identity information. In this method based on TTP model, TTP is the simplest intermediary entity between the user and the LBS provider. If the request is accepted, the request will be sent to the LBS provider; at the same time, the real ID will be changed to a pseudo-ID. In this way, the real ID is hidden for the provider. Even if the attacker obtains the accuracy location information, the exact interconnection between the user's location information and real ID still cannot be established. Through the pseudo-ID, the real ID could be concealed by user, which realizes the location privacy protection. Although this method can realize privacy protection to a certain extent, its shortage still exists. The server records all information of user's request and corresponding IP address, which will lead to the location privacy leak.

2.3. K-Anonymity Method. There is another location privacy protection method called k-anonymity method. Its idea comes from the k-anonymity model, which was proposed by Latanya Sweeney of the University of Carnegie Mellon in the United States. K-anonymity method was firstly proposed by Gruteser and Grunwald [16]. Before sending to the LBS provider, user deletes the personal information and publishes hypoaccurate data, which induces the fact that each record has identical quasi-identifier value with other $k - 1$ record in the data list. The identity of each user is accurately identified as $1/K$ under the condition of the same probability. K-anonymity method realizes the location privacy. But the restriction of k-anonymity method is that there is no protection mechanism for leak of sensitive attribute data, and there is not any constraint for sensitive attribute data in this method. It is easy for the attacker to infer the individual corresponding sensitive attribute data and identify the relationship between data and individual through the background information, which leads to the location privacy leak [17].

2.4. Personalized k-Anonymity Method and Other Methods. Because of the defect of k-anonymity method, other methods have been proposed to improve k-anonymity model. For example, A. Machanavajjhala proposed l-diversity model based on k-anonymity model. But this model is only suitable for dealing with classification sensitive attribute data instead of numerical sensitive attribute data. P-sensitive k-anonymity model could lose a lot of information usability in some data set and cannot resist the skewed attack and similarity attack to the sensitive attribute data. After P-sensitive k-anonymity model, (α, k)-anonymous model and (k, e)-anonymous model have the same defect. T-closeness frame can fix the skewed attack and similarity attack to the sensitive attribute data. But it reduces the usability of published data [18].

Personalized k-anonymity method was proposed by Gedik et al. [19, 20]. In this method, each user can define the desired anonymous level. The desired privacy level of every user is different, so personalized k-anonymity method is very popular. This method can provide different level of privacy protection to sensitive attribute data, which will decrease the data lost from the unified anonymous. But there is a defect in this method that the proportion of anonymous information will decrease when the K-value increases.

3. Improved K-Value Location Privacy Protection Method

In LBS, in order to achieve the protection, there are three main models: (1) noncooperative model; (2) peer-to-peer cooperative model; and (3) TTP model. This paper proposes the improved K-value location privacy protection method that is based on the TTP model, which will realize user location anonymous, service request anonymous, and feedback to the user. This model connects the user and LBS provider. Through the analysis, we found that all kinds of location privacy protection methods mentioned above based on TTP model have some disadvantages. For example, users need to customize the value of K when they use personalized k-anonymity method. But it is hard to choose the suitable value of K. The suitable value of K has a close connection with the quality of LBS service and the request of privacy protection. Therefore, we propose the improved K-value location privacy protection method based on the previous location privacy protection method.

3.1. Frame Model of Improved Method. Improved K-value location privacy protection method is a special location privacy protection method. In this system, the maximum and minimum values of K are needed to be set and there is no other default status. The location privacy level is obtained from the feedback learning of the ideal users. The value of K will be adaptive with the feedback and finally close to the requests of users. So we also call this method improved adaptive k-anonymity location privacy protection method. The frame model of the improved K-value location privacy protection method is shown in Figure 2.

3.2. Algorithms and Procedure of Improved Method

Algorithms. Set $k_{\min}$ as the minimum value of K and $k_{\max}$ as the maximum value of K; set $k_{\min} = 2$ and $k_{\max} = 8$.

(1) The user sends the service request to the server;

(2) The trusted third party receives the service request in the transmission process;

(3) Controller defines the level of privacy;

(4) Controller defines the value of K;

(5) If $K \in (k_{\min}, k_{\max})$;

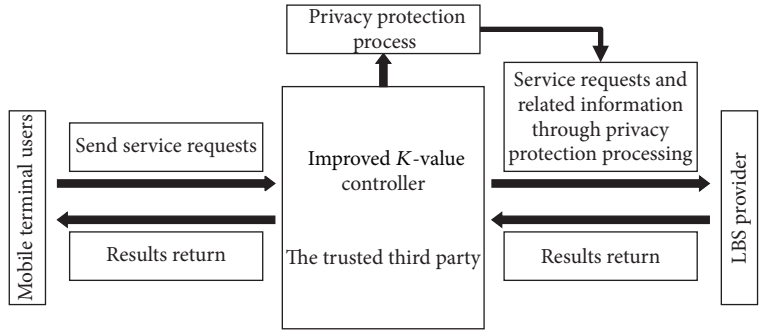


Figure 2: The frame model of the improved K-value location privacy protection method.

(6) The trusted third party processes the privacy protection with improved k-anonymity method and pseudonym method;

(7) Else;

(8) If $K < k_{\min}$;

(9) The trusted third party processes the privacy protection with k-anonymity method;

(10) Else;

(11) If $K > k_{\max}$;

(12) The trusted third party processes the privacy protection with pseudonym method;

(13) End if;

(14) End if;

(15) End if.

Summing up, the algorithm is summarized into a figure:

$$\text{process method} = \begin{cases} \text{improved } k\text{-anonymity method and pseudonym method}, & K \in (k_{\min}, k_{\max}) \\ k\text{-anonymity method}, & K < k_{\min} \\ \text{pseudonym method}, & K > k_{\max} \end{cases} \quad (1)$$

Procedure. From the above algorithms, algorithm procedure can be divided into three parts. (1) If the value of K is higher than $k_{\max}$, the trusted third party will adopt the pseudonym method to anonymize the user's location information. Here the real id will be replaced by the pseudo id and the pseudo id will be saved in the trusted third party database list with the real id and other detailed information of the user. When the trusted third party receives the result from the LBS provider and gets ready to send it to the mobile terminal, this system will check the corresponding real id of user in this list connected with the pseudo id, and then all primal data will be feedback to the mobile terminal user. (2) If the value of K is lower than $k_{\min}$, the trusted third party will adopt the k-anonymity method to anonymize the user's location information. The trusted third party will transmit the result to the LBS provider and send the feedback that received from the LBS provider to the mobile terminal user. (3) If the value of K is located in the set range, the trusted third party will adopt the k-anonymity and pseudonym methods to anonymize the user's location information. The trusted third party will send service request to the LBS provider, who will answer the request and return the results to the mobile terminal user.

In this algorithm, the range of values of $k_{\max}$ and $k_{\min}$ are as follows: the privacy disclosure threshold given by the data publisher is $P_{\max}$, the privacy disclosure probability of K-anonymity table is P, T is original data table, and T' is the K-anonymity table. The victim is U and the privacy attribute of value is S_u; each tuple with U is denoted as IG_u and $|\mathrm{IG}_u| = e$. The number of S_u that appears in IG_u is denoted as $|S_u| = f$; then the connecting candidate set of U is denoted as C_u and $|C_u| = g$.

Then, the probability of privacy disclosure of the individual attacked party U can be expressed as

$$P(U) = \frac{g^e - g^{e-f}(g-1)^f}{g^e} = 1 - \left(1 - \frac{1}{g}\right)^f. \quad (2)$$

In (2), $g^e - g^{e-f}(g-1)^f$ is the possible situation with some kind of special privacy and g^e is all. When the maximum number of replications of the sensitive attribute values in the tuple is l, according to $f \le l$ and $g \ge e \ge k$, then

$$P(U) = 1 - \left(1 - \frac{1}{g}\right)^f \le 1 - \left(1 - \frac{1}{k}\right)^l. \quad (3)$$

If $1 - (1 - 1/k)^l \le P_{\max}$, then $1 - (1 - 1/k)^l \ge 1 - P_{\max}$, and finally

$$k \ge \frac{1}{1 - (1 - P_{\max})^{1/l}}. \quad (4)$$

Therefore,

$$k_{\min} = \frac{1}{1 - \left(1 - P_{\max}\right)^{1/l}}. \quad (5)$$

Next, we use the identification metric C_{DM}. C_{DM} is represented as $C_{\mathrm{DM}} = \sum_{j=1}^{N} |\mathrm{IG}_j|^2$, where N is the number of tuples and $|\mathrm{IG}_j|$ is the scale of the j tuple in the anonymous table. Because $k \le |\mathrm{IG}_j| \le 2k - 1$, $k \le C_{\mathrm{DM}} \le 2k - 1$, when given $C_{\mathrm{DM}_{\max}}$ ($C_{\mathrm{DM}_{\max}} \ge 2k - 1$), then

$$k_{\max} \le \frac{C_{\mathrm{DM}_{\max}} + 1}{2}. \quad (6)$$

4. Experiment and Performance Analysis

Due to the limitation of the actual environment, we analyze and demonstrate the privacy protection methods through the simulation experiment.

4.1. Experimental Data Sets and Parameter Values. In this simulation experiment, we think that the automobiles along the road send the service request to the server, which are replaced by the moving object generators. The service request is based on the location information of the moving object generator. We use OPEN GL to simulate the national mapping map, which is provided by US Geological Survey. This map utilizes the spatial data transmission standards.

Experiment circumstance is Intel® Core™ i3-2310M 2.10 GHz for CPU, 6 GB for memory in Windows 10 Multiple Editions. Programming circumstance is Eclipse + Hibernate + SQL Server 2014. In this experiment, 500 moving object generators were used to simulate the automobile along the road and 580-service-request information was received. K-value was set as 2, 3, 4, 5, 6, 7, and 8.

4.2. Performance Analysis. In order to measure the results of the experiment, we adopt the anonymized success rate. The anonymized success rate is the ratio of the number of requests anonymized by the trusted third party to the total number of requests sent to the trusted third party, which is an important parameter for performance evaluation of privacy protection methods. It can reflect the response capability of the location privacy protection algorithm to the user's service request; the higher the value of the algorithm is, the better the capability will be.

4.3. Experiment Results. In the simulate experiment, we input different K-values in order to check the working of the algorithm in different contexts. With the value of K changing constantly, it is discovered that the number of information anonymized using different methods is different. When the value of K is small, we find that the most information is anonymized with k-anonymity method and only a bit of information is anonymized by pseudonym method. However, with the enlargement of K-value, more and more information will be anonymized with pseudonym method while less and less information will be anonymized by k-anonymity method.

Table 1: Number of information anonymized by k-anonymity method and pseudonym method.

K-value	2	3	4	5	6	7	8
The number of messages using k-anonymity method	503	432	360	293	222	156	72
The number of messages using pseudonym method	77	148	220	287	358	424	508

Table 2: Execution time of anonymous process using different methods.

K-value	2	3	4	5	6	7	8
The execution time of improved method	1.30 s	1.54 s	1.79 s	2.01 s	3.48 s	4.05 s	5.67 s
The execution time of personalized k-anonymity method	1.29 s	1.59 s	1.87 s	2.43 s	4.11 s	5.30 s	7.03 s

That is to say, when the value of K is less, the k-anonymity method will be used more and as the value of K increases, the use of pseudonym method will get increased. The number of information anonymized by k-anonymity method and pseudonym method is recorded in Table 1.

In order to reflect the efficiency of anonymous algorithm that we proposed, we record the execution time of anonymous process using different methods, which is the time of anonymous process for all inquiry requests from a certain scale of mobile uses. If the execution time is shorter, the anonymous algorithm is more efficient. Otherwise, the efficiency of the anonymous algorithm is worse.

In this experiment, we compare the execution time of improved K-value location privacy protection method and personalized k-anonymity method. From the results, it is discovered that when the value of K is small, the execution of time is almost the same for both methods. However, with the enlargement of the value of K, we realize that the execution time of personalized k-anonymity method is significantly longer than the execution time of the previous, which is owing to its much deeper refinement to the data and bigger searching space. In this execution of the personalized k-anonymity method, after each refinement, the restraint of every new anonymous group is needed to be calculated and the sensitive attribute generalization will also be undertaken by it, which costs longer execution time. The execution time of anonymous process using improved method and personalized k-anonymity method is recorded in Table 2.

Figures 3 and 4 are used to compare the results of the experiment more directly.

In Figure 3, we can see that the number of information anonymized by k-anonymity method will decrease with the enlargement of K-value, which is opposite to the number of information anonymized by pseudonym method.

From Figure 4, we see that the execution time of our proposed method is less than the personalized k-anonymity

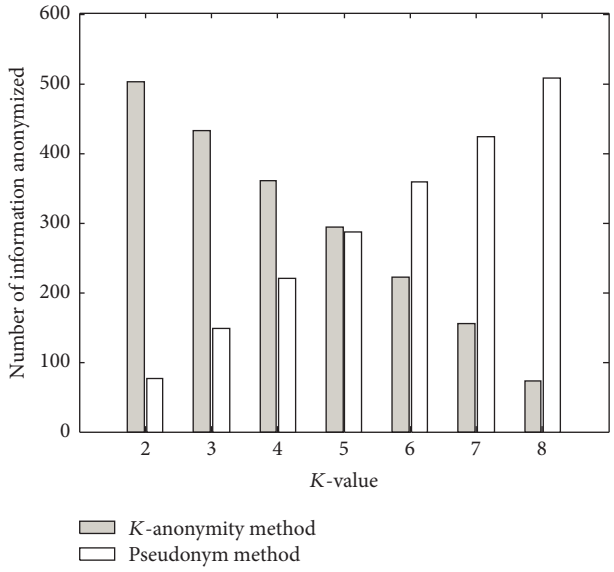


Figure 3: Number of information anonymized by k-anonymity method and pseudonym method.

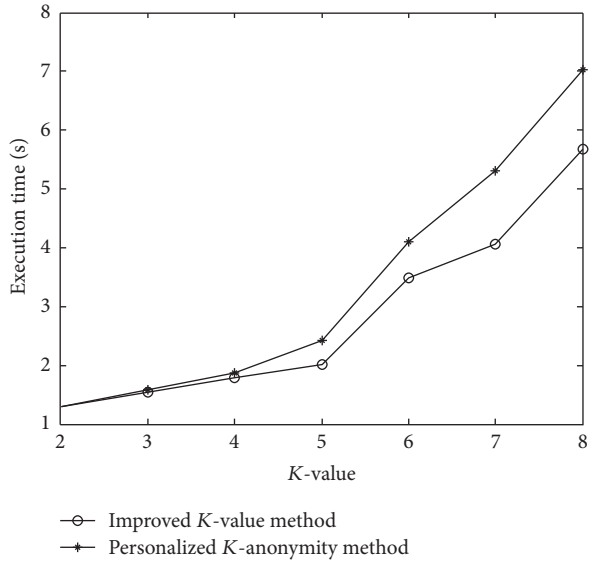


Figure 4: Execution time of anonymous process using different methods.

method, which means that the efficiency of this improved method is higher.

5. Conclusions

In this paper, we analyze the location privacy protection method in Augmented Reality, which is worth being paid attention to. It is crucial for privacy protection and data quality to choose reasonable K-values, so we propose the improved K-value location privacy protection method based on the previous methods. This method could define the value of K according to the level of privacy protection. Then different privacy protection methods are adapted according to the value of K to process the user's location privacy. Through the simulation experiment, the method we propose can anonymize all service requests effectively with shorter execution time, which realizes the location privacy protection more efficiently. However, the location privacy protection method is not perfect in Augmented Reality and it needs further study on the relevant issues.

Competing Interests

The authors declare that they do not have any conflict of interests related to this work.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (61373134). It was also supported by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD), Jiangsu Key Laboratory of Meteorological Observation and Information Processing (KDXS1105), and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

References

[1] P. Milgram and F. Kishino, "Taxonomy of mixed reality visual displays," *IEICE Transactions on Information and Systems*, vol. E77-D, no. 12, pp. 1321–1329, 1994.

[2] E. Olshannikova, A. Ometov, Y. Koucheryavy, and T. Olsson, "Visualizing Big Data with augmented and virtual reality: challenges and research agenda," *Journal of Big Data*, vol. 2, no. 1, pp. 1–27, 2015.

[3] Wikipedia, "Augmented reality," http://en.wikipedia.org/wiki/Augmented_reality.

[4] M. Deidda, A. Pala, and G. Vacca, "An example of a tourist location-based service (LBS) with open-source software," *Applied Geomatics*, vol. 5, no. 1, pp. 73–86, 2013.

[5] C. Yin and J. Xi, "Maximum entropy model for mobile text classification in cloud computing using improved information gain algorithm," *Multimedia Tools & Applications*, 2016.

[6] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, and S. Li, "Incremental support vector learning for ordinal regression," *IEEE Transactions on Neural Networks & Learning Systems*, vol. 26, no. 7, pp. 1403–1416, 2015.

[7] B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman, and S. Li, "Incremental learning for ν-support vector regression," *Neural Networks*, vol. 67, pp. 140–150, 2015.

[8] C. Yin, L. Feng, and L. Ma, "An improved Hoeffding-ID data-stream classification algorithm," *Journal of Supercomputing*, vol. 72, no. 7, pp. 2670–2681, 2016.

[9] C. Yin, L. Ma, and L. Feng, "A feature selection method for improved clonal algorithm towards intrusion detection," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 30, no. 5, Article ID 1659013, 2016.

[10] X. Li, M. Miao, H. Liu, J. Ma, and K.-C. Li, "An incentive mechanism for K-anonymity in LBS privacy protection based on credit mechanism," *Soft Computing*, 2016.

[11] I. Memon, "Authentication user's privacy: an integrating location privacy protection algorithm for secure moving objects in

location based services," *Wireless Personal Communications*, vol. 82, no. 3, pp. 1585–1600, 2015.

[12] Y. Sun, L. Yin, L. Liu, and S. Xin, "Toward inference attacks for k-anonymity," *Personal and Ubiquitous Computing*, vol. 18, no. 8, pp. 1871–1880, 2014.

[13] M. Bialke, P. Penndorf, T. Wegner et al., "A workflow-driven approach to integrate generic software modules in a Trusted Third Party," *Journal of Translational Medicine*, vol. 13, article 176, 2015.

[14] M. Uzielli, F. Catani, V. Tofani, and N. Casagli, "Risk analysis for the Ancona landslide—I: characterization of landslide kinematics," *Landslides*, vol. 12, no. 1, pp. 69–82, 2015.

[15] J.-H. Song, V. W. S. Wong, and V. C. M. Leung, "Wireless location privacy protection in vehicular Ad-Hoc networks," *Mobile Networks and Applications*, vol. 15, no. 1, pp. 160–171, 2010.

[16] B. Kenig and T. Tassa, "A practical approximation algorithm for optimal k-anonymity," *Data Mining and Knowledge Discovery*, vol. 25, no. 1, pp. 134–168, 2012.

[17] P. Belsis and G. Pantziou, "A k-anonymity privacy-preserving approach in wireless medical monitoring environments," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 61–74, 2014.

[18] B. Zhou and J. Pei, "The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks," *Knowledge and Information Systems*, vol. 28, no. 1, pp. 47–77, 2011.

[19] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: architecture and algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, 2008.

[20] Z. Xia, C. Yuan, X. Sun, R. Lv, D. Sun, and G. Gao, "Fingerprint liveness detection using difference co-occurrence matrix based texture features," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 11, no. 11, pp. 1–16, 2016.