2018

Efficient Information Dissemination in Vehicular Networks with Privacy Protection

Xiaolu Cheng
Virginia Commonwealth University

EFFICIENT INFORMATION DISSEMINATION IN VEHICULAR NETWORKS WITH PRIVACY PROTECTION

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

by

XIAOLU CHENG

B.E., Shandong University of Science and Technology, China - September 2009 to July 2013

M.S., Virginia Commonwealth University, USA - August 2014 to May 2016

Director: Wei Cheng, Ph.D.,

Assistant Professor, Department of Computer Science

Virginia Commonwealth University

Richmond, Virginia

August, 2018

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisor Dr. Wei Cheng. In the past four years, he supports my Ph.D. study and research with his patience and knowledge.

I would like to thank the rest of my committee members and all other professors who have taught me. I appreciate their patience, motivations, and encouragement.

I am grateful to all my labmates. Their kindness and intelligence make me entirely enjoy the four years with them.

I feel honored to be a Ph.D. student in the Virginia Commonwealth University Department of Computer Science. The department provides all my research needs and funds my research. Thank all faculty and staffs in the department for their help.

TABLE OF CONTENTS

Chapter Page

Acknowledgements i

Table of Contents ii

List of Tables v

List of Figures vi

Abstract ix

1 Introduction 1

1.1 Background 1
1.2 Motivation 1
1.3 Proposed Research 3

2 Related Work 6

2.1 Efficient Information Dissemination in VANETs 6
2.2 Security and Privacy Protection in VANETs 9
2.3 Other Related Topics 12
2.4 Uniqueness of our Works 13

Thrust I: Efficient Information Dissemination in VANETs 16

3 Task I.1: Roadside Unit Assistant Based Dissemination 16

3.1 Problem Statement 16
3.2 Basic Definitions and Meet-Table 17
3.3 Message Distribution with Meet-Table and TTL 20
3.3.1 Principle and Algorithms 21
3.3.2 Performance Analysis 22
3.4 Messages Distribution Based on Meet-Cloud 28
3.4.1 Principle and Algorithms 28
3.4.1.1 Definitions and Deployment 28
3.4.2 Architecture and Algorithms of Meet-Cloud 30
3.4.3 Message Distribution and Redistribution Algorithm 32

3.4.4 Performance and Security Analysis 32
3.4.4.1 Performance Analysis 33
3.4.4.2 Security Analysis 35
3.4.5 Simulation and Results 37
3.5 Messages Distribution Based on Meet-Fog 43
3.5.1 Architecture of Meet-Fog 43
3.5.2 Algorithms of Meet-Fog 45
3.5.3 Performance Analysis 46
3.6 Conclusion 50
4 Task I.2: Self-Organized Cluster-Based Dissemination 54
4.1 Problem Statement 54
4.2 Proposed Approach 58
4.2.1 Overview and Assumption 58
4.2.2 Cluster Formation 59
4.2.2.1 Cluster Head Selection 60
4.2.2.2 Cluster Maintenance and Reforming 62
4.3 Performance Evaluation 63
4.3.1 Simulation Parameter 63
4.3.2 Results and Analysis 64
4.4 Conclusion 66
Thrust II: Security and Privacy Protection in VANETs 74
5 Task II.1: Improving Security and Stability of AODV with Fuzzy Neural Network in VANETs 74
5.1 Problem Statement 74
5.2 Proposed Approach 77
5.2.1 Improving AODV Protocol with Fuzzy Neural Network . . . 77
5.2.1.1 Arithmetic Statement 77
5.2.1.2 Node Stability Metrics 77
5.2.1.3 Node Security Metrics 79
5.2.1.4 Fuzzy Neural Network 80
5.2.1.5 Optimizing the Fuzzy Neural Network with Genetic Simulated Annealing Algorithm 81
5.2.2 GSS-AODV Protocol Description 82
5.2.2.1 Routing Initiation 82
5.2.2.2 Routing Choice 84
5.2.2.3 Routing Maintenance 85

5.3 Performance Evaluation 86
5.3.1 Stability Experimental Results and Analysis 86
5.3.2 Security Experimental Results and Analysis 89
5.4 Conclusion 91

6 Task II.2: Multi-Antenna Based Information Privacy Protection for V2V Communications 94

6.1 Problem Statement 94
6.2 Proposed Approach 95
6.2.1 RSS-Ratio Properties Analysis 95
6.2.2 Anonymous Connection Setup and Maintenance 97
6.3 Conclusion 99

References 101

Vita 113

LIST OF TABLES

Table Page

1 Topic Summary 5

2 Parameters for Generating Simulation Dataset I 25

3 Summary of Performance Simulation Results I 25

4 Complexity of Distributing Method I 35

5 Message Coverage Metrics of Distributing Methods in Very Large VANET 35

6 Parameters for Generating Simulation Dataset II 38

7 Summary of Performance Simulation Results II 40

8 Message Coverage of Distribution Scheme 49

9 Complexity of Distributing Method II 50

LIST OF FIGURES

Figure Page

1 Principle of Message Distribution with Meet-Table and TTL 21

2 Braodcast_step Algorithm 23

3 AlgMeetTableTTL_Broadcast Algorithm 24

4 Time vs. Percent of Vehicles Met 24

5 Time vs. Percent of Vehicles Possessing Message 26

6 Time vs. Coverage Percentage 26

7 Time vs. Accurate Coverage Percentage 27

8 Message Percent vs. Time Delay 27

9 Alg-Aggregate Algorithm 29

10 Architecture of Scheme Based on Meet-Table and Cloud Computing . . . 30

11 Architecture of Meet-Cloud 31

12 Alg-Distribute Algorithm 33

13 Alg-ReDistribute Algorithm 34

14 Percent of Vehicles and RSUs Met on the Test Dataset 38

15 Performance of Different Distribution Methods 39

16 Performance of Meet-Cloud under Fake Meet-Table Attack 41

17 Performance of Meet-Cloud under DoS of RSUs 42

18 Architecture of Meet-Fog 47

19 Initiate-System Algorithm 47

20 Manage-RSU Algorithm 48

21 Manage-Vehicle Algorithm 52

22 Distribute-Message Algorithm 53

23 Redistribute-Message Algorithm 53

24 An Example of Vehicular Network 55

25 A Simplified View of Clustered Vehicular Networks 56

26 Communications within One Cluster 57

27 Clustering Algorithm 68

28 Cluster Head Selection Algorithm 69

29 Cluster Maintenance Algorithm 69

30 Average CH Lifetime VS. N and R 70

31 Average CH Lifetime VS. R and v 70

32 Average CM Lifetime VS. N and R 71

33 Average CH Lifetime VS. R and v 71

34 Average Number of Re-affiliation Times per Vehicle VS. N and R 72

35 Average Number of Re-affiliation Times per Vehicle VS. R and v 72

36 Packet Loss Rate VS. N and R 73

37 Packet Loss Rate VS. R and v 73

38 GSS-AODV Structure 77

39 Fuzzy Neural Network Structure 80

40 Routing Initiation Process Flow Diagram 82

41 Stability Experimental Results and Analysis 92

42 Security Experimental Results and Analysis 93

43 Impact of Transmission Power on RSS and RSS-Ratio 96

44 RSS-Ratio Distribution 97

45 Anonymous Connection Setup Algorithm 99

Abstract

EFFICIENT INFORMATION DISSEMINATION IN VEHICULAR NETWORKS WITH PRIVACY PROTECTION

By Xiaolu Cheng

A submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

Virginia Commonwealth University, 2018.

Director: Wei Cheng, Ph.D.,

Assistant Professor, Department of Computer Science

Vehicular ad hoc network (VANET) is a key component of Intelligent Transportation System (ITS). In VANETs, vehicles and roadside units exchange information for the purpose of navigation, safe driving, entertainment and so on. The high mobility of vehicles makes efficient and private communications in VANETs a big challenge.

Improving the performance of information dissemination while protecting data privacy is studied in this research. Meet-Table based information dissemination method is first proposed, so as to improve the information dissemination, and to efficiently distribute information via utilizing roadside units, Cloud Computing, and Fog Computing. A clustering algorithm is proposed as well, to improve the stability for self-organized cluster-based dissemination in VANETs on highways.

Then, fuzzy neural networks are used to improve the stability and security of routing protocols, AODV, and design a novel protocol, GSS-AODV. To further protect data privacy, a multi-antenna based information protection approach for vehicle-to-

vehicle(V2V) communications is also proposed.

CHAPTER 1

INTRODUCTION

1.1 Background

Intelligent Transportation System (ITS) takes a vital part in Smart Cities. ITS integrates information technology, data communication technology, electronic sensing technology, electronic control technology, computer processing technology, and some other advanced technology together into an intelligent transport management system[1].

ITS can be applied to not only vehicle systems but also railway systems, air transport system sand water transport systems[1]. ITS processes real-time information. Communication technology is an essential component of ITS.

In vehicular networks, there are two types of communications[2]. Vehicular networks enable both vehicle-to-vehicle (V2V) communications and vehicle-to-infrastructure (V2I) communications. In VANETs, vehicles and the infrastructures, such as Roadside Units (RSU) and application servers, exchange information for navigation, safe driving, entertainment and so on. Vehicular networks attract researchers from different fields, and massive research efforts have been made.

1.2 Motivation

After reading about three hundreds of papers in different fields of vehicular networks, such as application, protocol, architecture, and simulation, I would like to focus on the problems of data dissemination and network secure, which are the most fundamental and important parts in vehicular networks. I aim to improve the performance of data dissemination while protecting information privacy.

Among the data should be sent in vehicular networks, there are many important messages, such as CRL (Certificate Revocation List) and blacklist information. Unlike other data such as movies, these messages are very important to a particular group of vehicles which may encounter the vehicle relevant to the message. A vehicle is not able to know whom it will pass until they encounter. It may be too late if we wait for a vehicle requests for the message. If we use an epidemic dissemination method, the efficiency will be very low. On the other hand, the large quantity and high mobility of vehicles make the system very difficult to record and analyze the trajectories of them. To transmit these messages accurately and efficiently to the certain vehicles, we propose a series of RSU assistant information dissemination schemes.

Currently, communications in VANETs can via both Dedicated Short Range Communication (DSRC) and mobile cellular networks. To make use of existing mobile cellular networks for data transmissions, many methods are proposed to manage VANETs. However, if VANETs are fully managed by infrastructures, low efficiency will be a big issue, while, fully decentralized VANETs must create a lot of overhead. Grouping the vehicles into clusters and organizing the network by clusters is one of the most universal and most efficacious ways to solve this problem. Since the high mobility of vehicles makes VANETs different from other mobile ad hoc networks (MANETs), the previous cluster-based methods for MANETs may have trouble organizing VANETs. To achieve the goals of high-speed data transmissions and decreasing overhead, the clusters should be stable. Therefore, we propose a center-based clustering algorithm to help self-organized VANETs forming stable clusters and decrease the status change frequency of vehicles.

In networks, routing protocol decides which route the information should be sent to. Proposing a stable routing algorithm with security policy is essential. We aim to propose a routing protocol which is efficient, secure, and stable in unattended, harsh environments. Ad hoc On-Demand Distance Vector (AODV) Routing, as a routing protocol for wireless ad hoc networks, is widely used in VANETs. The original AODV Routing protocol assumes all nodes are not malicious, that is impractical for real VANETs. Therefore, we propose a secure and stable AODV named GSS-AODV, which uses a fuzzy neural network to compute the node information in routing activities. GSS-AODV uses trust value of the node to evaluate the node security. This evaluation balances node security with the network environment and node utilization to prevent malicious node attacks.

To further improve security and protect the privacy of vehicles, we also try to design a new kind of communication approach between vehicles. In the real world, vehicles meet each other occasionally. Drivers and passengers may just want to have a one-time conversation with the temporary neighbors. Users have the requirement of protecting its privacy. We, therefore, propose an idea of a multi-antenna based mechanism to protect a vehicle's real ID during communications.

1.3 Proposed Research

In this dissertation, we introduce proposed mechanisms in **Chapter 3** $\sim$ **Chapter 6** and summarize all tasks in Table 1.

In the first part, we focus on efficient information dissemination in **Chapter 3** and **Chapter 4** with two tasks.

In Task I.1, to improve information dissemination efficiency, we target on designing the data forwarding mechanisms for a special type of data via utilizing roadside units. Three Meet-Table based information dissemination methods are proposed to efficiently distribute information to a specified group of vehicles. The proposed data dissemination schemes are based on Meet-Table, Cloud Computing and Fog Computing. We expect it to yield high accuracy and efficiency.

In Task I.2, to improve the performance of communications in vehicular networks, we propose a stable clustering algorithm in VANETs on highways. It can send messages with high speed and low cost. The average frequency for vehicles to change their cluster status is low. Both the average Cluster Head (CH) lifetime and average Cluster Member (CM) lifetime are long. A novel cluster forming and Cluster Head (CH) selecting approach is designed for communications in VANETs on the highway. The Center-Based Stable Clustering Algorithm can help the vehicular networks to utilize DSRC resource as well as cellular resource and save the cost of communications in the whole system. In another word, with the help of Center-Based Stable Clustering Algorithm, the vehicular networks disseminates data efficiently.

In the second part, we focus on security and privacy protection in VANETs in **Chapter 5** and **Chapter 6**.

First, to improve security and stability of AODV in VANETs, we proposed a GSS-AODV Routing protocol in Task II.1. In GSS-AODV, a fuzzy neural network is employed to compute the node information in routing activities. The stability of nodes is computed to evaluate links. The link stability and the number of hops are considered in a balanced way, so a stable path with fewer hops is selected. GSS-AODV uses trust value of the node to evaluate the node security. The evaluation balances node security with network environment and node utilization to prevent malicious node attacks. In routing maintenance processes, GSS-AODV uses genetic simulated annealing algorithm to optimize the parameters of the fuzzy neural network in real time to ensure that the calculated stability and trust value of node match the actual situation.

An idea of the multi-antenna based information privacy protection approach to protect vehicle privacy in V2V communications is presented in Task II.2. This ap-

Table 1. Topic Summary

Task	Targeted Problems	Proposed Approaches
I.1	Roadside Unit Assistant Based Dissemination.	To a specified group of vehicles; New evaluation parameters; With the help of Meet-Table, Cloud Computing and Fog Computing.
I.2	Self-Organized Cluster Based Dissemination.	Center-based clustering algorithm; Stable clusters; Low overhead.
II.1	Improving Security and Stability of AODV in VANETs.	A stable routing algorithm with security policy; With the help of fuzzy neural networks.
II.2	Privacy Protections in V2V Communication	Without revealing IDs; Utilize RSS-Ratio.

proach utilizes the dynamic physical level information, RSS-Ratio, as the address for communications. A discussion on the feasibility of RSS-Ratio-based ID is presented in this chapter.

CHAPTER 2

RELATED WORK

This section is organized according to the corresponding related works for my proposed research topics.

2.1 Efficient Information Dissemination in VANETs

As a hybrid network, VANETs consists of three communication domains: in-vehicle domain, ad-hoc domain, and infrastructure domain[2]. The in-vehicle domain consists of On Board Unit (OBU) and Application Unit(OA), and is represented with a vehicle in this paper. The ad-hoc domain consists of vehicles. The infrastructure domain consists of vehicles and RSUs.

VANETs support many transportation applications to improve safety, efficiency, convenience, etc.[3]. The major goal of VANETs is to enhance the safety of transportation. To achieve this goal, vehicular networks must collect, process, and disseminate information, such as road conditions, position of the obstacles, speed limits, and road accidents, etc[3]. With the development of vehicular networks, especially when self-driving cars really run on the road, security of vehicular networks will be the key for safe transportation.

Secure vehicular networks require ID authentication, message integrity, communication confidentiality, guaranteed availability, and access control[4]. To meet these requirements, many solutions have been proposed. In these solutions, public key cryptography, trust management, blacklist, etc. are employed. Therefore, secure vehicular networks need to process messages about security in a secure and efficient way. As elsewhere, certificates used in vehicular networks must be revoked in circumstances, such as compromising or losing of a private key, illegal usage of a certificate, etc. [5].

CA (Certificate Authority) can issue CRL and store it on LDAP (Lightweight Directory Access Protocol) server for retrieving[6]. A vehicle can also use OCSP (Online Certificate Status Protocol) to request CRL[7]. Instead of directly accessing the Internet, vehicles in vehicular networks often access Internet through infrastructure domain, so both to retrieve CRL from LDAP server and to request CRL by using OCSP is not applicable.

Several broadcasting methods have been proposed for information dissemination, which can be used in vehicular networks. The authors of literature [8] propose a epidemic information dissemination system. The authors of literature [9] apply this epidemic information dissemination on vehicular networks and improve its efficiency. In [10], authors try to propagate CRL in an epidemic fashion. An epidemic method can distribute CRL to all vehicles with less number of RSUs and spend less time, but it requires large storage and high communication capacity in vehicular networks. Epidemic method is not economic. In [11], the authors improve and apply Dynamicity Aware Graph Relabelling Systems based on a tree-based topology management structure to vehicular networks. The Vehicle Infrastructure Integration tries to distribute CRL to vehicles through RSU broadcasting[12]. This method requires a very large number of RSUs and high cost.

The epidemic method can distribute CRL to all vehicles with less number of RSUs and less time, but it requires large storage and communication capacity in VANETs. Uncontrolled Epidemic may cause flooding storm, so TTL is often used. For example, TTL is used to restrict the number of message replicas that a node is allowed to spread[13]. In literature [14], TTL is used to limit the Hop-Tree update range to avoid over-overlapping of paths. But in VANETs, a large TTL can hardly restrict flooding, while a small TTL will sharply decrease the coverage of messages.

According to the research results in [15] [16] [17], a VANET is a Small world. In [18], a query processing algorithm that can determine the scope of each query is used to help a vehicle to avoid returning overwhelmed large amount results. These works give us a clue to accurately distribute messages in VANETs.

On the other hand, the framework of VANETs is very important. Ref. [19] proposes a method, named LTE4V2X, to organize vehicular networks. In the centralized vehicular networks, eNodeB manages vehicles in its coverage and divides them into clusters. LTE4V2X protocol defines how the self-organized network works. In LTE4V2X, eNodeB creates clusters which contain the largest number of nodes circulating in the same direction.

Ref. [20] extends LTE4V2X to increase information dissemination efficiency. It selects CHs by the distance from vehicles to eNodeB. Although compared to the original approach, the complexity is lower and the LTE channel quality is higher, the power consumption of message exchanging is not optimized. Nevertheless, [20] states that the system can calculate the transmit power of DSRC channels by the distance between vehicles so that the transmit power could be dynamically adjusted.

Road condition affects the speed and direction of vehicles. For example, vehicle's speed is lower on the bumpy road than on a smooth road. Vehicle mobility is determined by human behavior. Taking a street connected a megapolis and a village as an example, in the morning, most vehicles move from the village (home) to the megapolis (office); in the evening, most vehicles run following the reverse path. [21] quantifies temporal locality similarity to measure the relation of two vehicles' mobilities. Then, it utilizes the relation of vehicles' movements to form stable clusters.

Ref. [22] proposes an approach to minimize the total power consumed by DSRC communications. They use a weighted distance matrix to indicate power consumed between each pair of vehicles. In this way, the CH selection problem is formulated as a variant of the p-median problem in graph theory [23]. In this approach, the number of clusters p is determined first based on LTE coverage radius and DSRC coverage radius. The p cluster zones are determined by vehicle quantity and 802.11P coverage radius. p Cluster Heads that are closest to the eNodeB are selected. Then, the system dynamically selects new CHs to minimize the transmission power between CMs and CH based on weighted distance and the p-median issue in graph theory. Although this approach minimizes the power consumption within a single cluster, the power consumption of V2I communications has not been considered. The method to decide the zones is vague and complicated. Moreover, this approach is not suitable for the scenario that CMs not only send their information to CH but also communicate among themselves.

Ref. [24] proposes a high-integrity file transfer scheme for VANETs on highways named Cluster-based File Transfer Scheme (CFT). In this scheme, CMs help their CH to download file fragments, and then, transmit fragments to the CH which requests the file. Since the very high speed of vehicles on highways, CFT is a good approach to help the vehicles download files which they do not have enough connection time to download. However, CFT just considers the bi-direction environment. In addition, with CFT CH broadcasts its request to its neighbors; then, neighbors receive the invitation of joining the cluster and broadcast the request to invite more vehicles to join the cluster until there are enough vehicles. Therefore, CFT may not able to apply in complicated environment, and it may cause network congestions.

2.2 Security and Privacy Protection in VANETs

AODV is a typical on-demand routing protocol widely used in VANETs and plays an important role in the development of VANETs[25]. For VANETs, the most basic requirement is that the designed routing protocol is efficient, secure, and stable in unattended, harsh environments. It is very important to propose a stable routing algorithm with security policy.

At present for this problem, many domestic and foreign literature on the AODV protocol have been studied and improved. The literature [26] proposes an improved TAODV routing protocol based on trust mechanism to determine whether the node is a malicious node by comparing the trust value of nodes. The literature[27] uses a fixed time window to judge whether the node is selfish or not, and there is a delay in judging the behavior of the node. Although the protocol in the literature[28] can detect changes in node behavior, there is a problem of insufficient evidence in calculating the trust value.

The literature[29] puts forward that the TARF routing protocol uses a neighbor table to record the trust degree and energy consumption of each neighbor node and prevent attacks based on routing location. However, routing protocol increases routing load when broadcasting energy control packets. The literature [30] based on the AODV routing protocol, uses the public key to encrypt and identify IP addresses. However, encryption technology creates many communication, computation and memory costs in the key distribution process.

The literature [31] introduces CBM-AODV, which combines the success rate and the link quality, improves the path stability in the routing process and can effectively prevent the link failure. The literature[32] proposes the LLA method to find a stable communication path, which focuses on the improvement of multi-hop paths and link stability. It makes the routing meet the requirements of Quality of Service (QoS) and provide real-time security information services. The literature [33] proposes a reactive routing protocol AODVCS based on the biologically inspired cuckoo search algorithm. The protocol uses the Cuckoo Search Algorithm (CSA) to determine the shortest path between two nodes, and adds a route trust prediction mechanism while ensuring the complete routing security and reliability of the routing protocol packet delivery rate, end-to-end delay, and other performance.

We utilize Received Signal Strength (RSS) in Task II.2 to protect vehicle privacy in V2V communications.

RSS is widely used for security purpose. [34] utilizes the difference between the RSS value on different devices in a body area network to design an authentication mechanism. [35] uses the difference between RSS values to proximity to detect proximity device and achieves a reliable securely pairing scheme. [36] designs a secure method for mobile devices. The system named Wanda is able to help mobile devices joining a local network, pairing with each other and be configured in cloud.

Security and privacy are key issues in vehicular networks. To achieve a secure vehicular network, many works have been done. [37] introduces IEEE 1609.2 WAVE communication standards, which is a Dedicated Short Range Communications protocol, and the Vehicle Safety Communications Project. The authors propose a novel mechanism for RSU to transmission certificate revocation information and protect privacy. In [38], the authors analyze black hole attack in vehicular networks. In [39], the authors analyze Black Hole and Gray Hole Attack in vehicular networks and design a framework against Black Hole and Gray Hole Attacks. [40] discusses six attacks in vehicular networks: Message Suppression Attack, DoS Attack, SYN flooding Attack, Alteration Attack, Link Spoofing Attack and Link withholding Attack and Fabrication Attack. To defend against these attacks, authors propose a security scheme via using the techniques of Multiple Operating Channels, Pool of Registered Vehicles, and Directional Antenna. The results of simulation show that the security scheme can reduce the number of successful attacks.

Privacy protection takes an important role in vehicular networks. Using pseudonyms is one of the most common methods to achieve privacy. [41] designs a decentralized data validation approach to prevent attackers from getting private data from a centralized center. [42] proposes a context-based pseudonym change algorithm which can use vehicles' information to create pseudonyms. [43] groups vehicles and adds a random silent period in communication to obscure a target vehicle's position.

In [44], authors design a mechanism of pseudonym changing and updating. They also try to increase the strengthen unlinkability. [45] proposes privacy schemes to protect vehicles' privacy when vehicles are using navigation services. In [46], authors propose data a novel disseminating protocol to protect receivers' position privacy with the help of roadside units.

2.3 Other Related Topics

Radio Frequency Identification (RFID) technology has been widely used in vehicular networks.

[47] and [48] introduce RFID-based vehicle management systems for a community. These two systems have the function of controlling a vehicle. The system in [47] can calculate parking fee, and the system in [48] can detect vehicle theft and give an early warning. [49], [50], and [51] are about how to monitor vehicles by RFID. In [49], a positioning method is proposed. The authors design a Vehicle Operational Control system and a Level IV Intersection Control system.

[51] proposes a method to estimate vehicle speed. It analyzes vehicle position information and develops driving instructions to assist the vehicle driving. The system in [50] is used to track vehicles and can also reduce traffic congestion probability in some ways such as controlling the traffic signal. It analyzes the traffic situation with vehicle information and driving information.

[52] and [53] propose safety RFID-based preventive systems which can inform

the drivers the presence of pedestrians and the road oddities, respectively. Their ultimate goals are to ensure safe driving. Their working principles are very similar. In [52], the pedestrian carries an RFID tag while, in [53], an RFID tag is installed on the road with a particular distance before the road oddity. The vehicle is equipped with an RFID reader to detect pedestrians and oddities. [54] presents a method to make the non-intelligent vehicles be able to communicate with an intelligent system, disregarding it is a stand-alone system or a cooperative system.

Ref. [55] proposes a real-time RFID localization system whose average accuracy is 0.6 meters. [56] designs a distributed target location estimation model using quantized RSS data. [57] and [58] design weighted localization system in vehicular networks with the help of RSS span.

2.4 Uniqueness of our Works

Our Roadside Unit assistant based information dissemination approach proposed in Task I.1 is different from existing information dissemination approaches because it disseminates information to a specified group of vehicles efficiently with the help of Meet-Table. We disseminate information only to the vehicles which are interested in the information. There are a few works about distributing information such as CRL (Certificate Revocation List) in VANETs [59] [10]. In [59], The Vehicle Infrastructure Integration (VII) tries to distribute CRL to a vehicle through RSU broadcasting. This method requires a very large number of RSUs and high cost. In [10], Haas et al. try to propagate CRL in an epidemic fashion. The epidemic method can distribute CRL to all vehicles with less number of RSU and less time, but it requires the large storage and communication capacity in VANET. In our previous work, we propose Meet-Table to optimize CRL propagation in VANETs[60]. The Epidemic method[10] can rapidly distribute messages in VANET through broadcasting. But in a large VANET, if there is no limitation of broadcasting, flooding storm may destroy the availability of it. Therefor, we propose a series of information dissemination with the the help of Time-To-LiveTTL) to restrict flooding [13], Cloud-Computing and Fog-Computing.

Several cluster-based approaches are proposed to speed up information disseminating for managing vehicle networks. How to form clusters and how to select Cluster Head are the key issues. In Task I.2, we propose a novel clustering approach to form clusters and select Cluster Head. Unlike most existing approaches, our approach focuses on the stability of clusters. While increasing the stability of clusters, the communication overhead decreases and the quality of communications increases. Previous methods, e.g., [61] and [62], divide the vehicles by lanes or angle, which change frequently. Instead of the movement information, two bytes, which indicate the direction from their origin to destination, are used to divide the vehicles first. Therefore, the Cluster Members in one cluster have a lower possibility to run far away from the Cluster Head. The Center-Based Stable Clustering Algorithm also has some other uniqueness. The clustering method uses the densest area to decrease the number of clusters, and a new relative mobility metric is introduced to reduce the influence of vehicles type and drives' driving habits to the stability.

In Task II.1, unlike other proposed AODV protocol, we use fuzzy neural networks to solve the stability and security of routing protocols, the node trust value and node stability are obtained through fuzzy calculation. On this basis, a stable AODV protocol with security policy is proposed based on the original AODV protocol. The node trust value and the node stability are obtained through the fuzzy neural network when the route is initiated according to various influencing factors of the vehicle and finally applied to the routine activities of the protocol. Protocols improve link stability, save routing repair costs, reduce the impact of malicious nodes, and improve network security. The simulation results show that compared with the AODV protocol, this protocol improves network performance under different environments.

Task II.2 aims to protect vehicles' privacy. Most existing privacy protection schemes utilize cryptography and context information. We create temporary IDs with dynamic physical level information named RSS-Ratio. A remote malicious vehicle cannot get the vehicle's real ID. Moreover, our multi-antenna based information privacy protection approach uses three antennas and RSS-Ratio to eliminate other environmental unknowns. It has much higher accuracy than other existing RSS-based approaches.

Thrust I: Efficient Information Dissemination in VANETs

CHAPTER 3

TASK I.1: ROADSIDE UNIT ASSISTANT BASED DISSEMINATION

3.1 Problem Statement

In real VANETs, some nodes are not trustable. With the development of VANETs, dangerous and untrustworthy vehicle identification information can also play important role in safety application. These messages, such as dangerous driving, untrustworthy certificate, blacklist and so on, take a great role in VANETs. For example, there is a vehicle controlled by a malicious attacker. It broadcasts fake information to its neighbors. To protect other vehicles, its neighbors need to know this vehicle is in blacklist. The previous data dissemination approaches can not disseminate such information efficiently and accurately, since vehicles do not send a request for such information and the system does not know which node cares about the information.

Information management in vehicular networks has been studied in literature [3], and there are a few works about distributing CRL (Certificate Revocation List) in vehicular networks. Literature [10] tries to propagate CRL in an epidemic fashion. An epidemic method can distribute CRL to all vehicles with less number of RSUs and less time, but it requires significant storage and communication capacity in vehicular networks.

The previous data dissemination approaches can not disseminate information efficiently and accurately to the vehicles who care about the information since vehicles do not send a request for such information. Moreover, broadcasting the message to all vehicles is infeasible and high-cost. Therefore, we propose a novel information dissemination mechanism to solve the problem: how to send such safety information to the vehicles which may need it. Our work is based on the basic concept of Meet-Table, which is introduced in literature [60] first.

In our work, we focus on the message describes a vehicle's negative characteristic, such as very unusual movement, malicious behavior, and invalid certification. This approach also can be applied to other types of messages.

Since we do not exactly know which vehicle is interested in the message, we can not only use the speed of information delivery to evaluate a method. Accordingly, two parameters, coverage percentage and accurate coverage percentage, are defined to assess the performance of our approach.

3.2 Basic Definitions and Meet-Table

In this chapter, a message uniquely binds to an objective vehicle. As the message is not usable for all vehicles in the VANETs, it has a set of vehicles that may care about it. Formally, a message is

$$m \overset{\text{def}}{=} < o, d, C > \tag{3.1}$$

Where m is the negative message, and it is a 3-tuple consisted by o, d and C; o is the objective vehicle of m; d is the data in the message describing o; C is a set of vehicles that concern the message m.

The messages m is often generated by authority. The authority generally put several messages together and sign them as a whole document. We call this type of document as a message document. Formally, a message document

$$D \overset{\text{def}}{=} < M, a > \tag{3.2}$$

$$M = \{m_i | 0 \leq i \leq n_M\} \tag{3.3}$$

Where D is the message document, M is the set of messages, a is the authority, n_M is the number of elements in M.

As there are various objective vehicles in a message document, to distribute the message document is more complicated and difficult than to distribute messages separately.

For convenience of description, we define the set of vehicles and RSUs.

$$V = \{v_i | 0 \leq i \leq n_V\} \tag{3.4}$$

$$U = \{u_i | 0 \leq i \leq n_U\} \tag{3.5}$$

V is the set of all vehicles, n_V is the number of vehicles. U is the set of all RSUs, n_U is the number of RSUs.

In VANETs, we should process a message m in a way that: (1) can push m to all vehicle $v \in C$ as soon as possible; (2) for every vehicle $v \in C$, can get m with high availability. General message dissemination methods in VANETs try to distribute data to all vehicles. These methods are not very efficient and suitable for messages contain one vehicle's information. For example, broadcasting CRL in national wide VANETs is not only unfeasible but also unnecessary[60]. For evaluating the method processing messages, possessing percentage, coverage percentage, and accurate coverage percentage are used in our task.

Definition 3-1 The possessing percentage of a message is the percent of vehicles possessing the message in all vehicles. Formally, coverage percentage

$$r_p = \frac{|B|}{|V|} \tag{3.6}$$

where B is the set of vehicles that possess the message.

Definition 3-2 The coverage percentage of a message is the percent of vehicles possessing the message in vehicles concerning the message. Formally, coverage percentage

$$r_c = \frac{|B \cap C|}{|C|} = \frac{\sum_{b \in B} \begin{cases} 1 & b \in C \\ 0 & b \notin C \end{cases}}{|C|} \tag{3.7}$$

where B is the set of vehicles that possess the message, C is a set of vehicles that concern the message.

Definition 3-3 The accurate coverage percentage of a message is the percent of vehicles concerning the message in vehicles possessing the message. Formally, accurate coverage percentage

$$r_{ac} = \frac{|B \cap C|}{|B|} = \frac{\sum_{b \in B} \begin{cases} 1 & b \in C \\ 0 & b \notin C \end{cases}}{|B|} \tag{3.8}$$

where B is the set of vehicles that possess the message, C is a set of vehicles that concern the message.

The coverage percentage r_c represents the availability of the message, while the accurate coverage percentage r_{ac} of the message represents the efficiency of distribution method. Consequently, an evaluation criteria is:

Evaluation-Criteria-1: A good message distributing method should has both high coverage percentage r_c and high accurate coverage percentage r_{ac}.

According to **Evaluation-Criteria-1**, an ideal model of distributing message m is to make C, the set of vehicles concerning m, equals to B, the set of vehicles possessing m. General methods of disseminating information in VANETs try to broadcast m to all vehicles, its coverage percentage $r_c \rightarrow 100\%$, but its accurate coverage percentage $r_{ac} \rightarrow 0\%$. These methods are not applicable in the message distribution described. So, in this chapter, we propose a series of improved schemes for distributing messages to a specified group with the help of Meet-Table. Our final goal is to design a high-performance message distributing method which has both high coverage percentage and high accurate coverage percentage.

Accordingly, we need a method to get C, the set of vehicles concerning message m. In fact, vehicles in C are these vehicles that may encounter the objective vehicle o of message m. According to the reproducible moving patterns of human[63], we can assume that these vehicles passed by the same RSU may encounter each other. We can record vehicles passing an RSU or a vehicle with the table, called Meet-Table.

Formally, Meet-Table of w, an RSU or a vehicle, can be defined as

$$T_w = \{p_i | 1 \leq i \leq n_{T_w}\} \tag{3.9}$$

$$p_i \overset{\text{def}}{=} < v, t, c >, v \ passed \ w \tag{3.10}$$

$$T = T_i | o \leq i \leq n_U \tag{3.11}$$

where T_w is the Meet-Table generated by w; n_{T_w} is the number of elements in T_w; p_i is the ith record in T_w, that is a 3-tuple consisted of v, t, and c; v is a vehicle passed w c times by time t. T is the set of all Meet-Tables in RSUs.

3.3 Message Distribution with Meet-Table and TTL

The Epidemic method[10] can rapidly distribute messages in VANETs through broadcasting. But in a large VANET, if there is no limitation of broadcasting, flooding storm may destroy the availability of it. TTL (Time-To-Live) can be used to restrict flooding[13]. A large TTL can hardly restrict flooding, while a small TTL will sharply decrease the coverage of messages.

A Meet-Table of a vehicle is used to record vehicles the vehicle met. Therefore, we can reset TTL during the message broadcasting with the help of Meet-Table. Using Meet-Table with TTL to broadcast messages in VANETs, a balance between flooding control and message coverage can be easily achieved. In this section, we utilize Meet-Table to lead resetting of TTL in messages broadcasting in VANETs to achieve high availability and high coverage of the messages, illustrate the principle and give the algorithm of using Meet-Table with TTL in messages broadcasting in VANETs, and simulate Meet-Table with TTL in messages broadcasting and give the results.

3.3.1 Principle and Algorithms

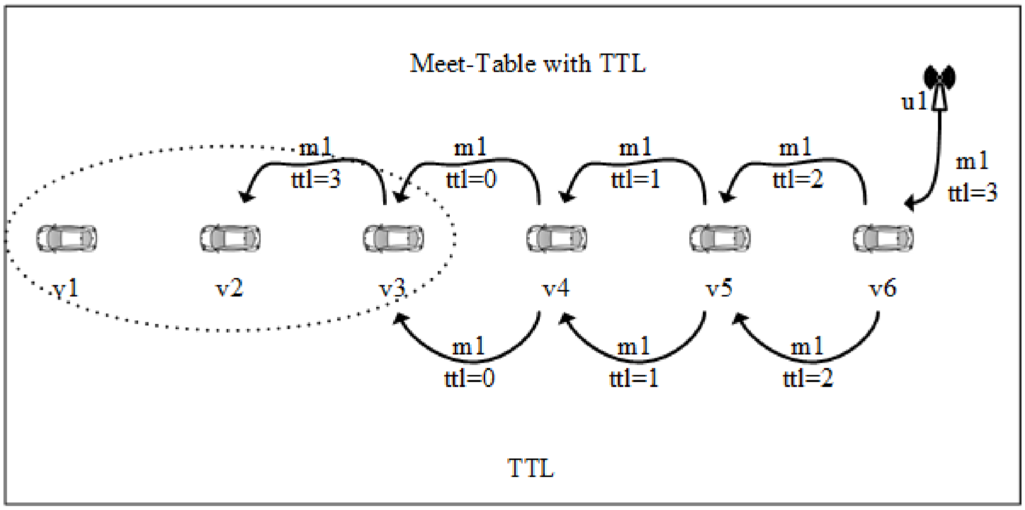


Fig. 1. Principle of Message Distribution with Meet-Table and TTL

Fig. 1 shows a sample VANET. In this VANET,

$$R = \{u_1\} \tag{3.12}$$

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6\} \tag{3.13}$$

$$m_1.o = v_1 \tag{3.14}$$

$$m_1.C = \{v_2, v_3\} \tag{3.15}$$

$$T_{v_3} = \{< v_1, ... >, ...\} \tag{3.16}$$

$$T_{v_2} = \{< v_1, ... >, ...\} \tag{3.17}$$

$$T_{v_1} = \{< v_2, ... >, < v_3, ... >, ...\} \tag{3.18}$$

$$TTL = 3 \tag{3.19}$$

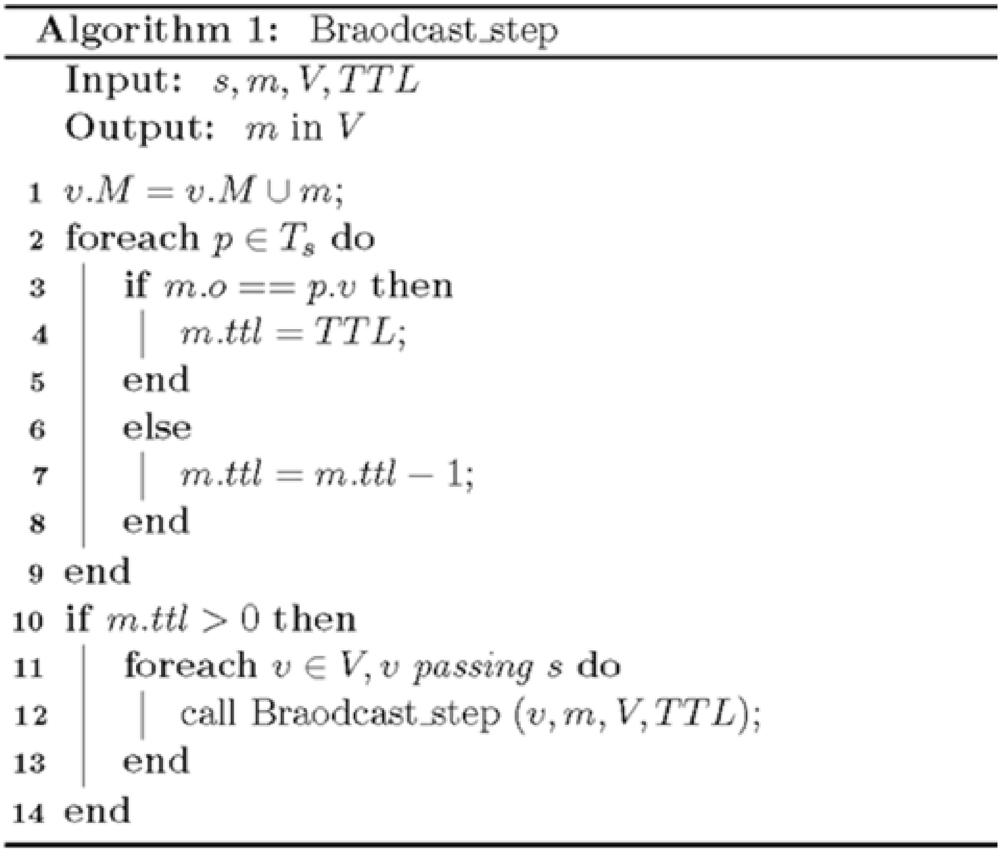
From Fig.1 we can see the enhancement of Meet-Table with TTL in messages broadcasting in the VANET. When TTL is used to control broadcasting, m_1 can only transmit to v_6, v_5, v_4 and v_3. v_2 cares m_1 but it can not get the message m_1.

When Meet-Table is used with TTL, m_1's ttl can be reset at v_3 for m_1's objective vehicle is v_1 and v_1 is in v_3's Meet-Table. Consequently, m_1 can be continually transmitted to v_2.

The algorithms using Meet-Table with TTL to enhance the messages broadcasting in VANETs are indicated in Fig. 2 and Fig. 3. In Broadcast_step Algorithm, s is a vehicle as the start point of broadcasting. m is the message waiting to be broadcasted. Obviously, this algorithm is recursive. It is called by itself and AlgMeetTableTTL_Broadcast. AlgMeetTableTTL_Broadcast Algorithm is the main algorithm. It takes m and V as inputs, generates a set of start points for broadcasting, sets TTL of m, then calls Broadcast_step Algorithm for all start points to fire the broadcasting.

3.3.2 Performance Analysis

To evaluate the performance of the proposed scheme, we generated a dataset to simulate all vehicles in San Francisco, USA.



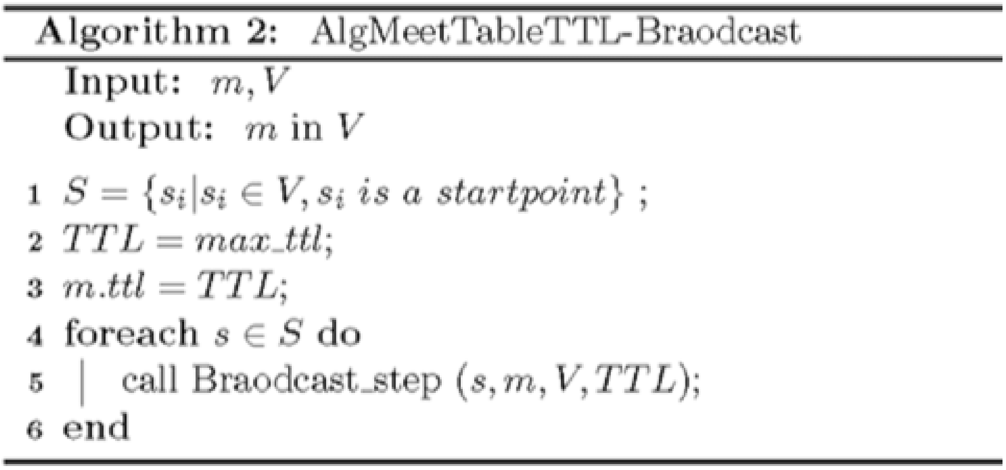
```
Algorithm 1: Braodcast_step
  Input: $s, m, V, TTL$
  Output: $m$ in $V$
  $v.M = v.M \cup m$;
  foreach $p \in T_s$ do
      if $m.o == p.v$ then
          $m.ttl = TTL$;
      end
      else
          $m.ttl = m.ttl - 1$;
      end
  end
  if $m.ttl > 0$ then
      foreach $v \in V, v$ passing $s$ do
          call Braodcast_step ($v, m, V, TTL$);
      end
  end
```

Fig. 2. Braodcast_step Algorithm

Simulation of VANETs can use the dataset of realistic traces of vehicles[15] or generate traces based on a map[10][64]. Realistic traces dataset of numerous vehicles is very hard to get. The dataset used in [15] is realistic Taxi GPS traces from Shenzhen and Beijing, China, and San Francisco, USA. The total number of vehicles in this dataset is only about 13000, and it only contains Taxi, no other types of vehicles. In addition, the time length of this dataset is no more than three days.

To evaluate the performance of the proposed scheme, we generated a dataset to simulate all vehicles in San Francisco, USA. The dataset was created based on parameters shown in Table 2.

Fig. 4 shows the percent of vehicles a vehicle met. The percent increases at



```
Algorithm 2: AlgMeetTableTTL-Braodcast
Input: $m, V$
Output: $m$ in $V$
$S = \{s_i | s_i \in V, s_i\ is\ a\ startpoint\}$ ;
$TTL = max\_ttl$;
$m.ttl = TTL$;
foreach $s \in S$ do
    call Braodcast_step $(s, m, V, TTL)$;
end
```

Fig. 3. AlgMeetTableTTL_Broadcast Algorithm

first, but keeps static after a certain point of time. This means that vehicles in the dataset have local attribute. In other words, the behavior of the vehicles like what demonstrated in literature[63].

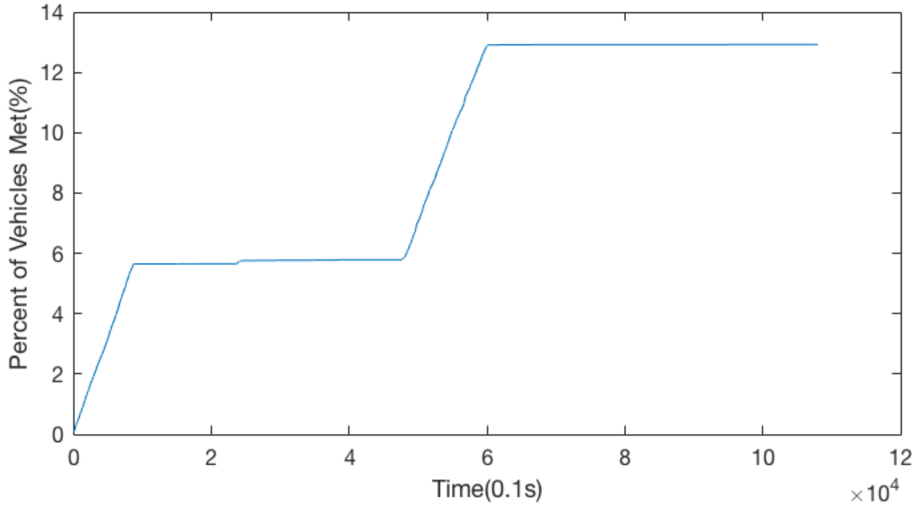


Fig. 4. Time vs. Percent of Vehicles Met

We simulate broadcasting methods of Epidemic without broadcasting control (shortly Epidemic), the Epidemic with TTL, and the Epidemic with both TTL and

Table 2. Parameters for Generating Simulation Dataset I

Parameter	**Value**	**Note**
Number of Vehicles	471388	Total number of vehicles in [65]
Intersections	7200	Estimated No. of Intersections in [65]
Length of road	1741(km)	Total length of road in [65]
Area	1 21(km^2)	Area Land in [65]
Mean Travel Time	0.5(h)	Mean Travel Time to Work in [65]
Speed	38.6(km/h)	Average speed of commuter traffic speeds in [66]
MaxV2I	100(m)	Max communication distance of vehicle to RSU
Start Points	0.0255%	
TTL	6	

Meet-Table (Epidemic with TTL and Meet-Table) on the generated dataset. The simulation results are shown from Fig. 5 to Fig. 8.

The summary of the simulation results is in Table 3.

Table 3. Summary of Performance Simulation Results I

Method Name	r_p	r_c	r_{ac}	**Delay**
Epdemic	high	High	Mid	Low
Epdemic with TTL	Low	Low	Low	High
Epdemic with TTL and Meet-Table	Min	High	High	Low

From Table 3. we can see that: (1) Meet-Table with TTL increases the coverage of messages; (2) Meet-Table with TTL has a higher accurate coverage percentage than Epidemic with and without TTL have; (3) The delay of Meet-Table is as low as that of Epidemic. So, Meet-Table with TTL can not only increase coverage but also

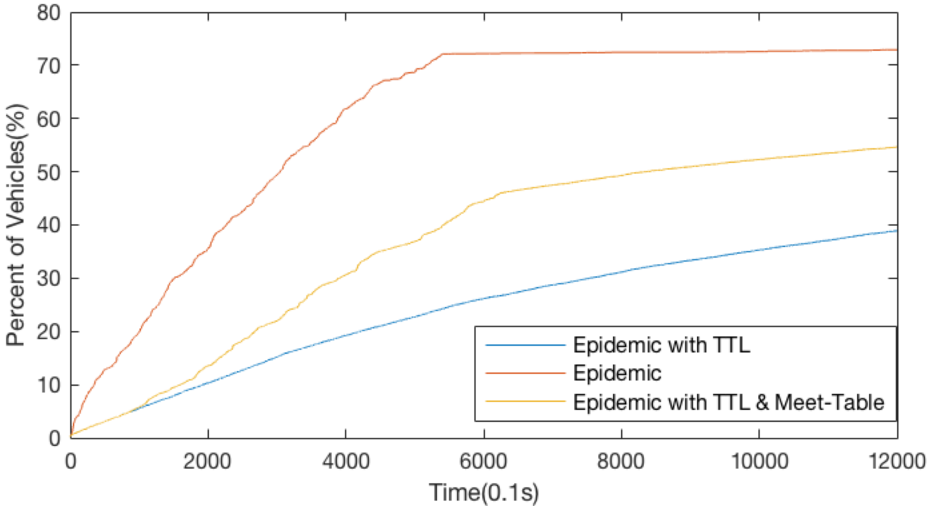


Fig. 5. Time vs. Percent of Vehicles Possessing Message

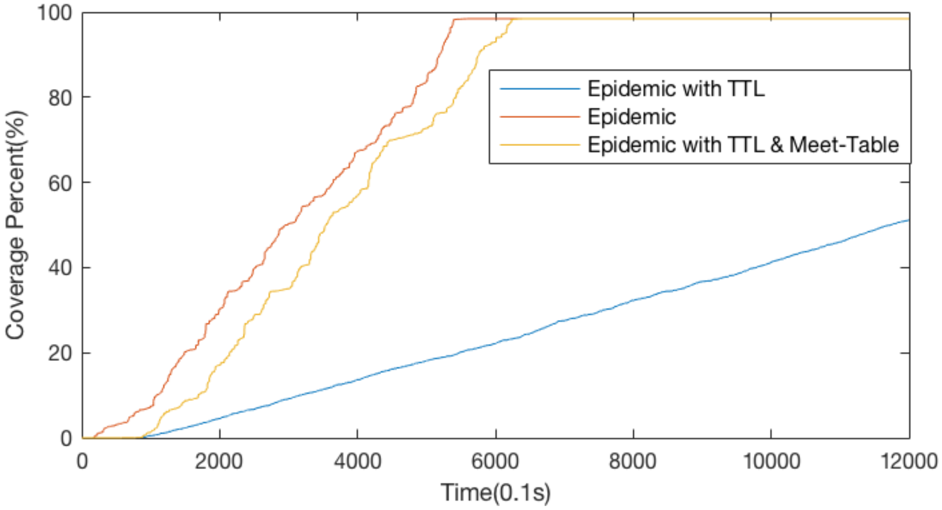


Fig. 6. Time vs. Coverage Percentage

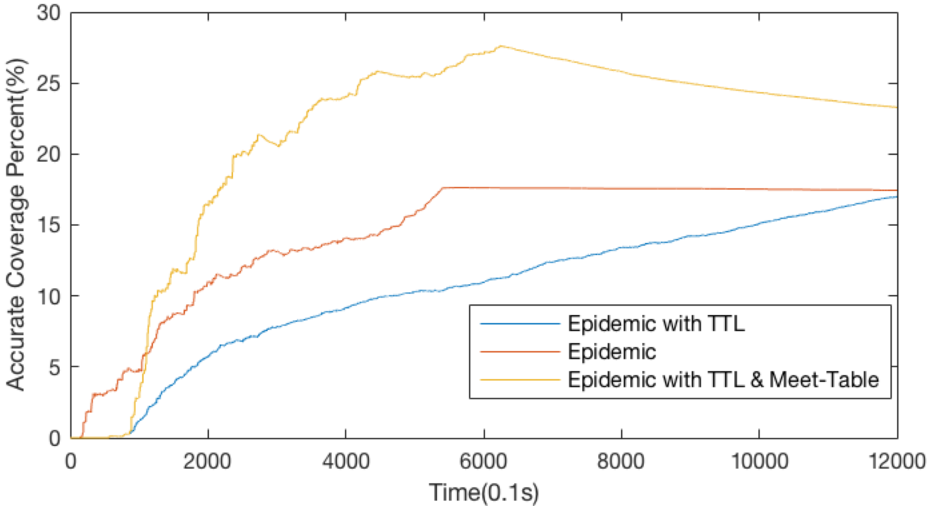


Fig. 7. Time vs. Accurate Coverage Percentage

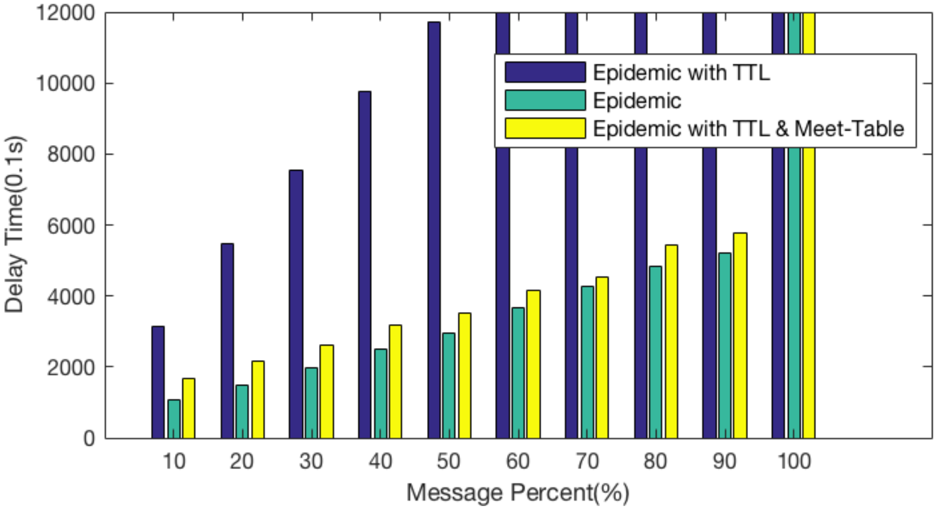


Fig. 8. Message Percent vs. Time Delay

accurately and timely increase coverage.

According to **Evaluation-criteria-1**, Epidemic with Meet-Table and TTL is better than Epidemic with or without TTL for distributing messages in VANETs.

3.4 Messages Distribution Based on Meet-Cloud

In previous section, we propose a data dissemination scheme with TTL to restrict flooding. To further improve information dissemination efficiency, we apply Cloud Computing to optimize Meet-Table based scheme.

3.4.1 Principle and Algorithms

3.4.1.1 Definitions and Deployment

In Section 3.3, Meet-Tables are distributed in RSUs. In this work, we construct a global Meet-Table for message distribution. Meet-Tables must be aggregated to a global form. Formally, the global Meet-Table can be defined as

$$G = \{g_i | 1 \leq i \leq n_G\} \tag{3.20}$$

$$g_i \overset{\text{def}}{=} < v, U_i > \tag{3.21}$$

where, G is the global Meet-Table; n_G is the number of elements in G. g_i is the ith recorder of G. v is the vehicle passed all RSUs in U_i.

The algorithm for aggregate Meet-Tables T to global Meet-Table G is presented in Alg-Aggregate presented in Fig. 9.

In a large VANET, the size of G may be huge, and its recorders have variable lengths, so it should be processed with NoSQL database [67] in Cloud Computing environment. Hence, we propose Meet-Cloud to make use of Meet-Table and Cloud Computing technology.

Algorithm 1: Alg-Aggregate

```
Input: $T, V, U$
Output: $G$
$G = \phi$ ;
foreach $v \in V$ do
    $U_v = \phi$ ;
    foreach $T_u \in T$ do
        foreach $p \in T_u$ do
            if $p.v == v$ then
                $U_v = U_v \cup \{u\}$ ;
            end
        end
    end
    $g =< v, U_v >$ ;
    $G = G \cup g$ ;
end
```

Fig. 9. Alg-Aggregate Algorithm

With the help of Meet-Table and Cloud Computing, we can efficiently distribute messages in VANET. The deployment of components in Meet-Cloud is shown in Fig. 10.

In Meet-Cloud, a Cloud Service is running on the Internet to process global Meet-Table and help to distribute negative messages. It utilizes high scalability and virtualization of Cloud Computing[68] and NoSQL Database to serve global Meet-Table processing and negative messages distributing. RSUs are built at the roadsides. They are connected to the Internet through wired or wireless communication channels, e.g. 5G[69]. Every RSU can record the vehicles passed it into its Meet-Table.

The Meet-Table of an RSU can be sent to the Cloud Service in a planned schedule. When an RSU receives a message from the Cloud Service, it broadcasts the message to those vehicles passing it.

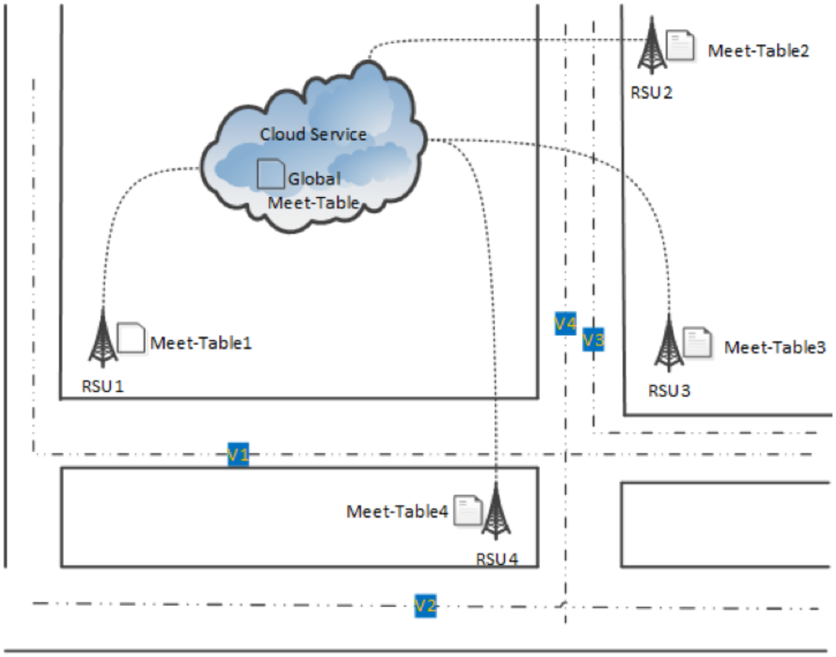


Fig. 10. Architecture of Scheme Based on Meet-Table and Cloud Computing

A vehicle travels along its ways. When it passes an RSU on the roadside, it can be recorded by the RSU. At the same time, it accepts messages broadcast by the RSU. If it comes across other vehicles, it can record them into its Meet-Table, and broadcast the messages gotten from RSUs it passed to them.

3.4.2 Architecture and Algorithms of Meet-Cloud

The principle of Meet-Cloud shown in Fig. 10 can be illustrated with the architecture shown in Fig. 11.

In the Meet-Cloud shown in Fig. 11,

$$V = \{v_1, v_2, v_3, v_4\} \tag{3.22}$$

$$U = \{u_1, u_2, u_3, u_4\} \tag{3.23}$$

$$T_{u_1} = \{< v_1, ... >\} \tag{3.24}$$

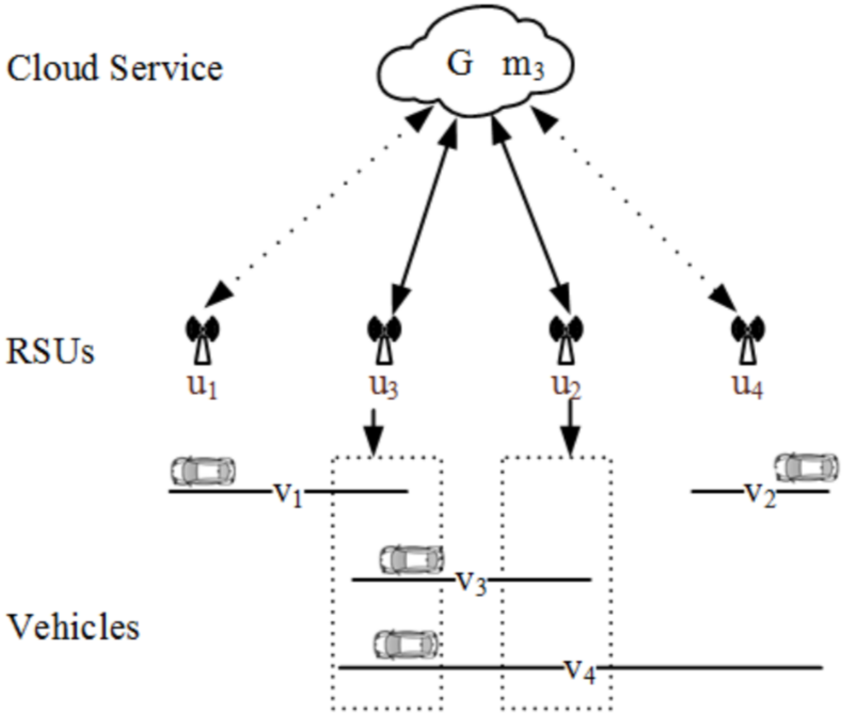


Fig. 11. Architecture of Meet-Cloud

$$T_{u_2} = \{< v_3, ... >, < v_4, ... >\} \tag{3.25}$$

$$T_{u_3} = \{< v_1, ... >, < v_3, ... >, < v_4, ... >\} \tag{3.26}$$

$$T_{u_4} = \{< v_2, ... >\} \tag{3.27}$$

$$G = \{< v_1, \{u_1, u_3\} >, < v_2, \{u_4\} >, < v_3, \{u_2, u_3\} >, < v_4, \{u_2, u_3, u_4 >\}\} \tag{3.28}$$

$$m_3 \overset{\text{def}}{=} < v_3, d, \{v_1, v_4\} > \tag{3.29}$$

So when m_3 is sent to the Cloud Service to distribute, it can find $< u_3, \{u_2, u_3\} >$ from G, and it can send m_3 to u_2 and u_3 for broadcasting. u_1 and u_4 does not need to do broadcast at all. Therefore, v_1 and v_4 will receive m_3, but v_2 will not.

In VANETs, if we know C, which is the set of vehicles that care the message m, we can accurately distribute m to vehicles in C. In fact, vehicles in C are these vehicles that may encounter the objective vehicle of m According to the reproducible moving patterns of human [63], we can assume that these vehicles pass by the same RSU may encounter each other. So, we can record vehicles passed an RSU or vehicle with Meet-Table. For message m, if $\exists p_i \in T_w$, $p_i.v = m.o$, then m should be distributed through w.

3.4.3 Message Distribution and Redistribution Algorithm

A message, which describes an attribute of its objective vehicle, is often distributed by an authorized entity. For example, CRL is a typical message issued by CA (Certificate Authority). In the proposed Meet-Cloud scheme, the algorithm for distributing negative message is presented in Alg-Distribute shown in Fig. 12.

Alg-Distribute is invoked by the entity that wants to distribute the message m, and executed by the Cloud Service, RSUs, and vehicles in an asynchronous and distributed model.

When an RSU u encounters a vehicle v that never encountered before, the RSU must redistribute negative messages of the vehicle to keep high coverage percentage and accurate coverage percentage of the messages. The message redistribution algorithm is presented in Alg-Redistribute Algorithm shown in Fig. 13.

Alg-Redistribute Algorithmis invoked by RSUs, and executed by RSUs. Vehicles in an asynchronous and distributed model. Every RSU executes its own Alg-redistribute procedure respectively. The Cloud Service provides the interface for querying messages of a vehicle.

3.4.4 Performance and Security Analysis

Meet-Cloud utilizes Meet-Table and Cloud Computing to securely and accurately distributing messages in VANET. We compare it with other methods to study its

```
Algorithm 2: Alg-Distribute
Input: $G, m$
Output: $m$ to $u, v$ where $u \in U, v \in V$
$U_m = \phi$ ;
foreach $g_i \in G$ do
    if $g_i.v = m.o$ then
        $U_m = g_i.U_i$ ;
        break ;
    end
end
if $U_m \neq \phi$ then
    foreach $u \in U_m$ do
        push $m$ to $u$ ;
        foreach $v$, which is passing $u$ do
            $u$ broadcasts $m$ to $v$ ;
            foreach $vv$, which comes across $v$ do
                $v$ broadcasts $m$ to $vv$ ;
            end
        end
    end
end
```

Fig. 12. Alg-Distribute Algorithm

performance formally, and analyze its security. There two typical methods, RSU broadcast[59] and Epidemic model[10], are used in VANETs to distribute messages. So we compare complexity and coverage of them.

3.4.4.1 Performance Analysis

For simplicity, we define several average quantities in VANETs. The average number of RSUs a vehicle may encounter

$$\overline{n_u} = \frac{\sum_{u \in U} n_{T_u}}{n_V} \tag{3.30}$$

```
Algorithm 3: Alg-ReDistribute
Input: $v, T_u$
Output: messages of $v$ to vehicles passing $u$
$p = null$ ;
foreach $p_i \in T_u$ do
    if $p_i.v = v$ then
        $p = p_i$ ;
        break ;
    end
end
if $p \neq null$ then
    $p =< v, current\_time, 1 >$ ;
    $T_u = T_u \cup \{p\}$ ;
    query $m, m.o = v$ from the Cloud Service ;
    if $m \neq \phi$ then
        foreach vv, which comes across u do
            broadcasts $m$ to $vv$ ;
            foreach $vvv$, which comes across $vv$ do
                $vv$ broadcasts $m$ to $vvv$ ;
            end
        end
    end
end
```

Fig. 13. Alg-ReDistribute Algorithm

The Average number of vehicles a RSU may encounter

$$\overline{n_u} = \frac{\sum_{u \in U} n_{T_u}}{n_U} \tag{3.31}$$

Then we can calculate complexities of Meet-Table, RSU Broadcast and Epidemic. The results are shown in Table 4.

From Table 4 we can see that: (1) Meet-Cloud can reduce communication from core to RSU and RSU to vehicle, for $\overline{n_u}$ is smaller than n_U. (2) Meet-Cloud can reduce communication between vehicles, for $\overline{n_u}$ and $\overline{n_v}$ are smaller than n_V. (3) Meet-Cloud can reduce vehicle storage, for $\overline{n_u} \cdot \overline{n_v}$ are smaller than n_V in a large VANET.

Table 4. Complexity of Distributing Method I

Complexity	Meet-Cloud	Epidemic Model	RSU Broadcasting
Core to RSU Communication	$n_M\overline{n_u}$	N/A	$n_M n_U$
RSU to Vehicle Communication	$n_M\overline{n_u} \cdot \overline{n_v}$	N/A	$n_M n_U \overline{n_v}$
Vehicle to Vehicle Communication	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M n_V^2$	N/A
Core Storage	$n_M + n_v\overline{n_u}$	N/A	N/A
Vehicle Storage	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M n_V$	$n_M n_V$
Computing	$n_U n_V$	N/A	N/A

RSU Broadcast and Epidemic try to distribute messages to all vehicles, but Meet-Cloud tries to distribute messages to the right vehicles that really care the message. In a very large VANET, message coverage metrics of these methods are shown in Table 5.

From Table 5 we can see that RSU Broadcast and Epidemic are not so efficient. According to **Evaluation-Criteria-1** and Table 5, Meet-Table is the best one.

Table 5. Message Coverage Metrics of Distributing Methods in Very Large VANET

Metric	Meet-Cloud	Epidemic	RSU Broadcast
Possessing Percentage	→ 0%	→ 100%	→ 100%
Coverage Percentage	→ 100%	→ 100%	→ 100%
Accurate Coverage Percentage	→ 100%	→ 0%	→ 0%

3.4.4.2 Security Analysis

We also describe the attack model and analyze the security of Meet-Table. Simulation results of Fake Meet-Table attack and DoS attack are given.

In the proposed scheme, we assume that authorized entity, Cloud Service, most

RSUs, and most vehicles are trustworthy. Under this assumption, we can profile the major attacks that can be conducted on the scheme.

(1) Fake message attack. An attacker tries to distribute the untrue message of a target vehicle to disturb communication and operation of the victim.

(2) Holding on message attack. An attacker tries to let vehicles received messages from RSUs do not broadcast the message to other vehicles encountered.

(3) Fake Meet-Table attack. An attacker tries to build fake Meet-Table by driving vehicle to pass lots of RSUs that are not necessary to pass in a normal human travel model.

(4) DoS (Denial of Service) attack. An attacker tries to jam broadcasting of RSUs, to block messages pushed from cloud service, to stop Cloud Service, to broadcast a huge number of garbage messages, etc.

From the architecture and the algorithms described above, we know that the proposed scheme executes in a distributed and asynchronous model, so the scheme has some potential anti-attack properties. Also, utilizing the matured Cloud Computing technology, the Cloud Service is scale free and hard to attack. There are lots of anti-attack measurements for fake message attack. For example, Cloud Service can authenticate the sender; and messages may be signed with signature for verification in RSUs and vehicles.

If a vehicle is controlled by an attacker, it may not broadcast messages received from RSUs and other vehicles to the vehicles it encounters. This holding message attack can hardly affect the propagation of negative messages in VANET, for comparing to other uncontrolled vehicles, the number of vehicles controlled by the attacker is very less.

An attacker can drive the vehicle passing RSUs to build fake Meet-Table, but it is very costly and easy to detect. This physical attack is hard to take place in a large scale. Besides, the movement pattern of the attackers vehicle is very different from the ordinary humans reproducible pattern[63], it is very easy to detect and clear them from global Meet-Table.

Generally, DoS, especially DDoS (Distributed DoS) is hard to defeat if opposite has enough resources[70]. In the proposed scheme, DoS, even DDos is hard to achieve its goal. If an attacker wants to jam broadcasting of an RSU, he/she must be at the site of the RSU, so he/she can only attack very limited RSUs. Because of the matured protect technology of Cloud Computing, it is difficult for the attacker to block messages pushed from Cloud Service or stop Cloud Service. An attacker can broadcast a huge number of garbage messages to a limited part of VANETs and affect a limited area of it, but he/she ca not affect the whole VANET, even the main part of it, for it is distributed, executed asynchronously, and has numerous RSUs and vehicles.

In summary, the scheme is secure to face these four types of attacks if it is implemented carefully, as it is distributed, executed asynchronously, has numerous entities, and is based on Cloud Computing technology.

3.4.5 Simulation and Results

After analysis of Meet-Cloud, we simulate it and other message distributing methods, and study the performances of them. Additionally, we simulate Meet-Cloud under DoS attack of RSUs and fake Meet-Table attack. The results are given and analyzed in this section.

Simulation of VANETs can use the dataset of realistic traces of vehicles[15] or generated traces based on a map[10][64]. Realistic traces dataset of numerous vehicles are very hard to get. The dataset used in [15] is realistic Taxi GPS traces from Shenzhen and Beijing, China, and San Francisco, USA. The total number of vehicles

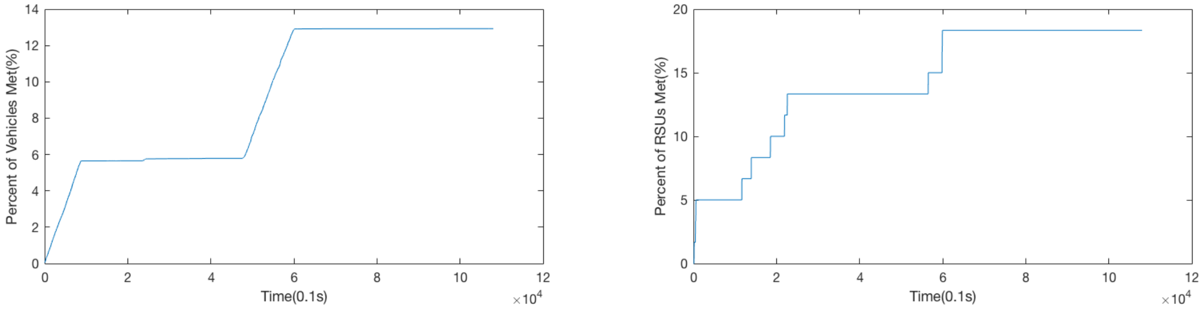
Table 6. Parameters for Generating Simulation Dataset II

Parameter	**Value**	**Note**
Number of Vehicles	471388	Total number of vehicles in [65]
Number of RSUs	1193	Refer to the No. of Signalized Intersections in [65]
Intersections	7200	Estimated No. of Intersections in [65]
Length of road	$1741(km)$	Total length of road in [65]
Area	$1\ 21(km^2)$	Area Land in [65]
Mean Travel Time	$0.5(h)$	Mean Travel Time to Work in [65]
Speed	$38.6(km/h)$	Average speed of commuter traffic speeds in [66]
MaxV2I	$100(m)$	Max communication distance of vehicle to RSU
MaxV2V	$10(m)$	Max communication distance of vehicle to vehicle

in this dataset is only about 13000, and it only contains Taxi, no other types of vehicles. In addition, the time length of this dataset is no more than three days.

To evaluate the performance and anti-attack ability of the proposed Meet-Cloud, we generated a dataset to simulate all vehicles in San Francisco, USA. The dataset was created based on parameters shown in Table 6.

On the generated dataset, the percent of vehicles and RSUs a vehicle met versus time are shown in Fig. 14 (a) and Fig. 14 (b) respectively.



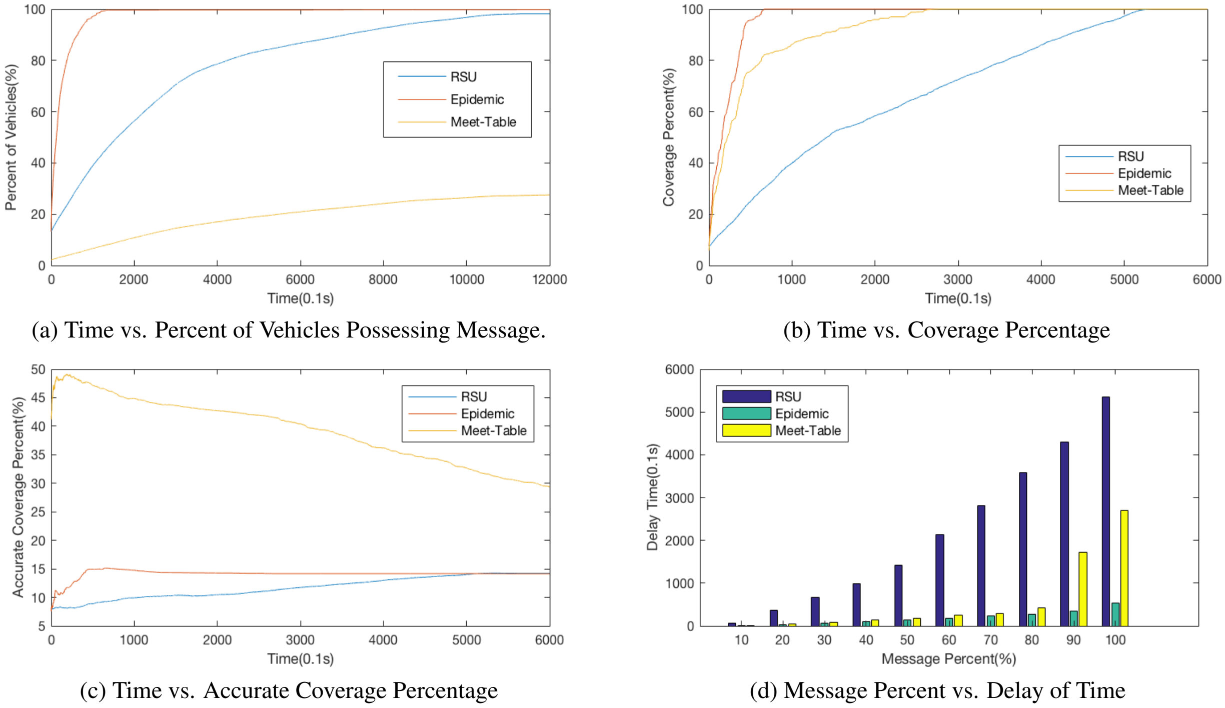
(a) Time vs. Percent of Vehicles Met

(b) Time vs. Percent of RSUs Met

Fig. 14. Percent of Vehicles and RSUs Met on the Test Dataset

From Fig. 14 we can see that both the percents of vehicles and RSUs met keep increasing at first, but keep static after a point of time. This pattern represents the locality of vehicles movement. So the generated dataset has the same attribute of movement of human in real daily life[15].

To compare the performance of the proposed Meet- Cloud with RSU broadcasting and epidemic model, we simulate these three methods on the generated dataset. The simulation results are shown in Fig. 15.



(a) Time vs. Percent of Vehicles Possessing Message.

(b) Time vs. Coverage Percentage

(c) Time vs. Accurate Coverage Percentage

(d) Message Percent vs. Delay of Time

Fig. 15. Performance of Different Distribution Methods

Table 7 summaries the performance simulation results.

According to Table 7 and Evaluation-Criteria-1, for the proposed Meet-Cloud has both high coverage percentage and high accurate coverage percentage, and mid message delay, it should be the best method for distributing messages between the three methods in VANETs.

Table 7. Summary of Performance Simulation Results II

Value Name	Epidemic Model	RSU Broadcasting	Meet-Table Based Scheme
Percent of Vehicles Possessing Message	Very high	High	Low
Coverage Percentage	Very high	High	High
Accurate Coverage Percentage	Low	Low	High
Message Delay	Low	High	Mid

In order to study the performance of Meet-Cloud under fake Meet-Table attack, we randomly add records into Meet-Table of RSUs and do simulation. The performances of different ratio of fake Meet-Table records are shown in Fig. 16.

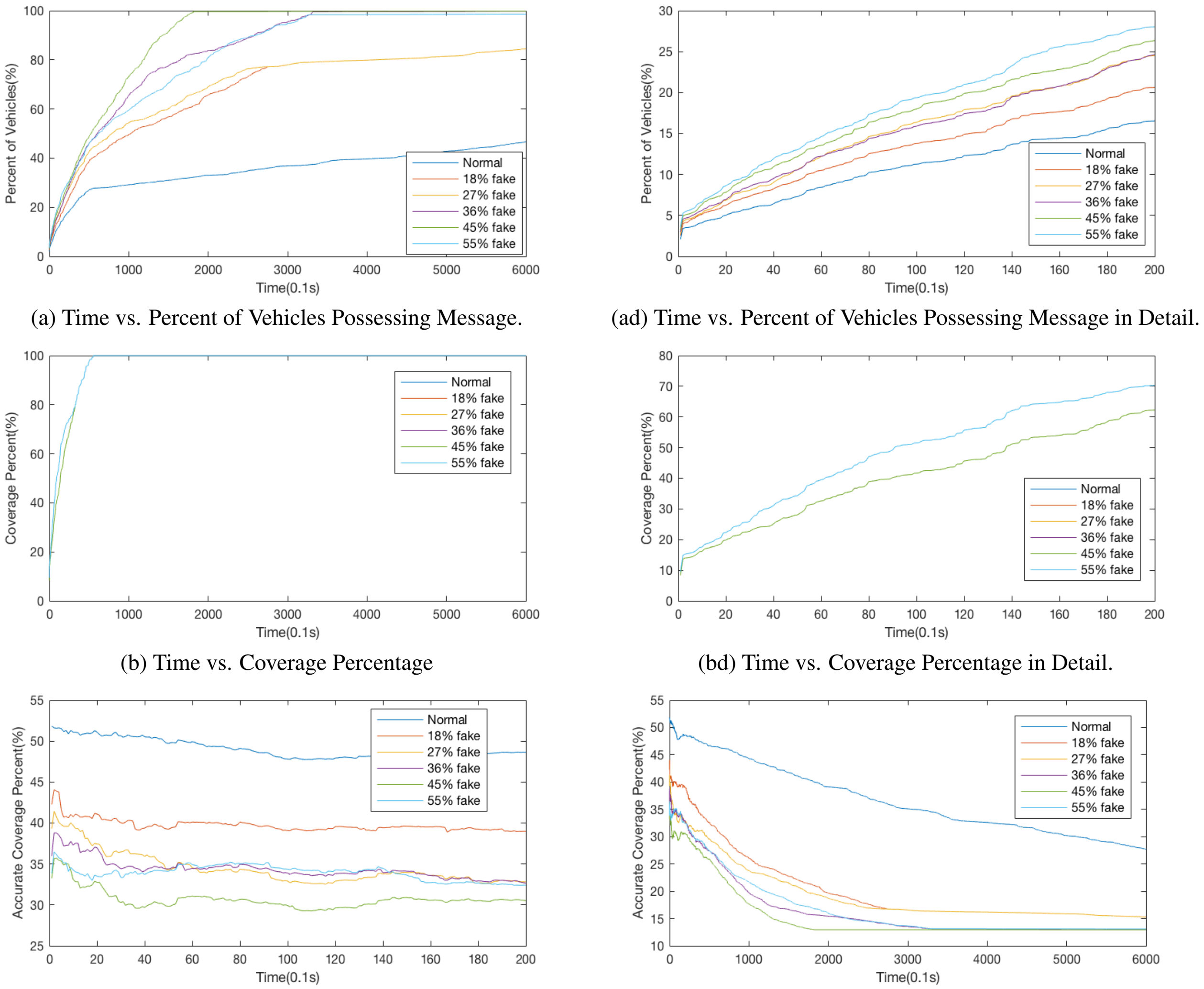
From Fig. 16 we can see that fake Meet-Table leads Meet-Cloud to act like Epidemic, and the higher ratio of fake Meet-Table records there is, the more Epidemic the Meet-Cloud goes to be like. But fake Meet-Table attack can only leads to low accurate coverage percentage, not low coverage percentage. In other words, fake Meet-Table attack can only affect the accuracy of message distributing, not range of message distributing. So Meet-Cloud is secure under fake Meet-Table attack.

We randomly turn off RSUs to simulate DoS attack of RSUs. Performance of Meet-Cloud with different ratio of RSUs off are shown in Fig. 17.

From Fig. 17 we can see that Dos of RSUs cant heavily affect performance of Meet-Cloud. So the proposed Meet-Cloud is secure facing DoS attack.

In fake Meet-Table attack we found that fake records in Meet-Table may lead Meet-Cloud to act like Epidemic. In DoS attack we found that turning off RSUs can not heavily affect the performance of Meet-Cloud. These attributes of Meet-Cloud make it secure to face these attacks.

But there is a fact we noticed. That is the proposed Meet-Cloud has attributes of both RSU broadcast and Epidemic. In fact, it uses the Meet-Table to select several RSUs as the starting point of Epidemic. So when there is a high ratio of fake Meet-



(a) Time vs. Percent of Vehicles Possessing Message.

(ad) Time vs. Percent of Vehicles Possessing Message in Detail.

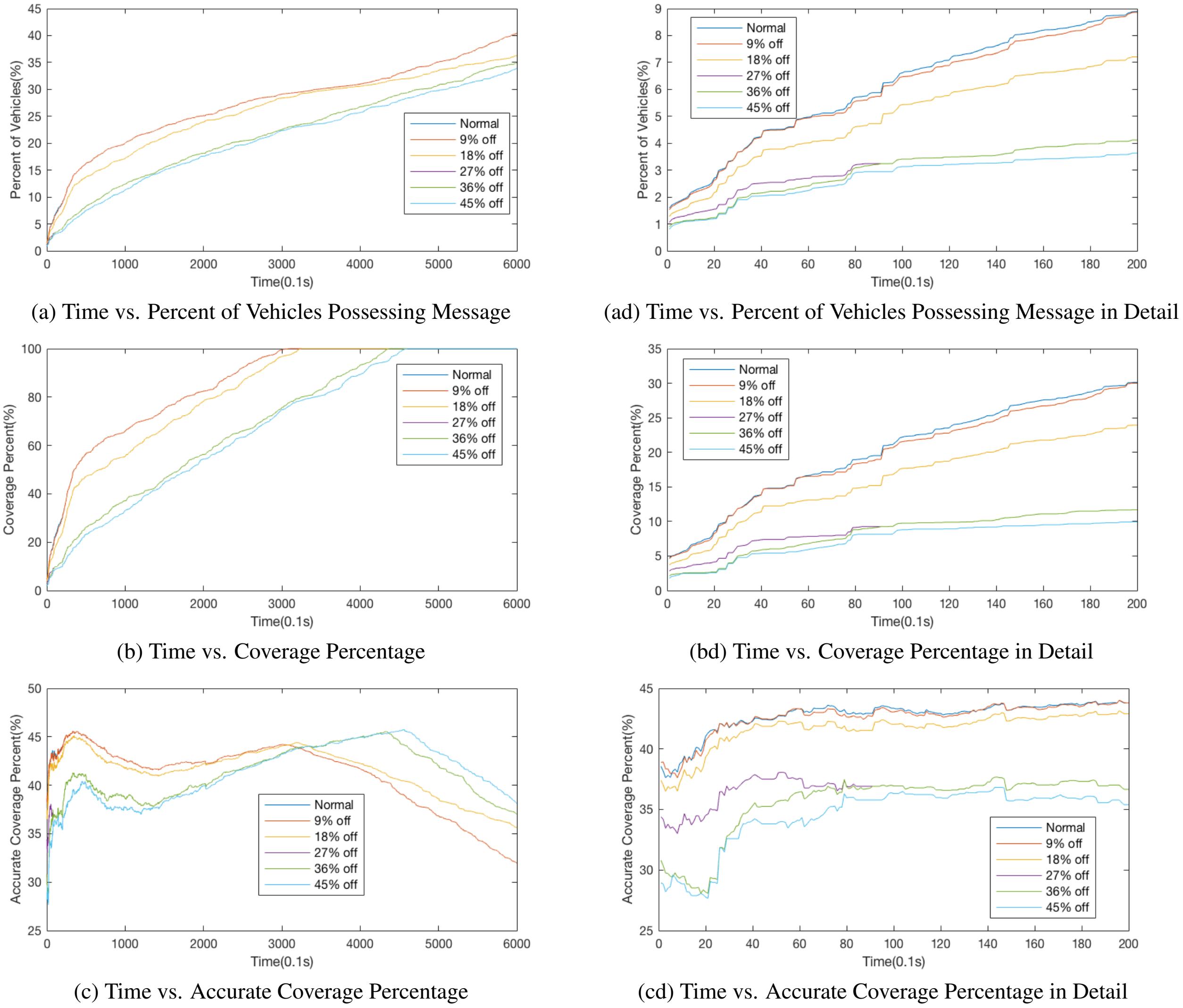
(b) Time vs. Coverage Percentage

(bd) Time vs. Coverage Percentage in Detail.

(c) Time vs. Accurate Coverage Percentage

(cd) Time vs. Accurate Coverage Percentage in Detail.

Fig. 16. Performance of Meet-Cloud under Fake Meet-Table Attack



(a) Time vs. Percent of Vehicles Possessing Message

(ad) Time vs. Percent of Vehicles Possessing Message in Detail

(b) Time vs. Coverage Percentage

(bd) Time vs. Coverage Percentage in Detail

(c) Time vs. Accurate Coverage Percentage

(cd) Time vs. Accurate Coverage Percentage in Detail

Fig. 17. Performance of Meet-Cloud under DoS of RSUs

Table records, the Epidemic attribute of Meet-Cloud is enhanced, and it act like Epidemic model. When DoS of RSUs occurs, some RSUs will be off, and starting points of Epidemic will be less. But Epidemic model has a exponential distributing ability. If there is only one starting point, Epidemic can distribute messages around the point rapidly.

3.5 Messages Distribution Based on Meet-Fog

In the Cloud Computing based scheme, all RSUs report their Meet-Table to the cloud service, so as to generate global Meet-Table, and the cloud service must push every message to all RSUs that concerns it. So the cloud service needs not only significant computing and communication capability, but also huge storage capacity. Fog Computing can extend the Cloud Computing process to the edge of the network, it enjoys the characteristics of low latency and location awareness[71], so we propose Meet-Fog, a Meet-Table and Fog Computing based scheme, to utilize Fog Computing and optimize our previous Meet-Table and Cloud Computing based scheme by reducing bandwidth and storage requirements of cloud, and moving computing requirement from cloud to the edge.

3.5.1 Architecture of Meet-Fog

The architecture of Meet-Cloud is presented in the previous section. To optimize computing, communicating, and storage of Meet-Cloud, Fog Computing can be adopted. We can organize RSUs as Fogs to do a part of these tasks that must be done by Cloud service. We call this scheme Meet-Fog. The architecture of Meet-Fog is shown in Fig. 18.

In Meet-Cloud system, all RSUs send their Meet-Tables to Cloud Service, then Cloud Service generates global Meet-Table G. Cloud Service sends the message m_2

about v_2 to RSU u_2 and u_3, since v_2s moving range is covered by u_2 and u_3. In Meet-Cloud, Cloud Service must receive and store all Meet-Tables of all RSUs and generate global Meet-Table from these Meet-Tables, so Cloud Service needs communication and storage capacity, and very powerful computing capability. Additionally, the message is transmitted from Cloud Service to RSUs redundantly.

In Meet-Fog system showed in Fig. 18, RSU u_2 and u_3 are organized as a Fog to serve message distribution of v_2, and u_2 is selected as the head of the Fog. Then the record in global Meet-Table about v_2 can be generated in the Fog and sent to Cloud Service through the Fog head. For Cloud Service, messages about v_2 only need to be sent to the Fog head.

The architecture of Meet-Fog can only show the main idea. A formal model of it can make algorithm description and analysis of it more precise and easier.

A message m describes one vehicle's attribute is already defined at the beginning of this chapter. All messages in the VANET can be denoted by a set

$$M_u = \{m_i | 0 \leq i \leq n_m\} \tag{3.32}$$

In order to organize RSUs as Fog, we must define the set of neighbor RSUs of an RSU and extend Meet-Table definition.

The set of neighbor RSUs of RSU u

$$H_u = \{u_i | u_i \in U, u_i \neq u, u_i \leftrightarrow u\} \tag{3.33}$$

$u_i \leftrightarrow u$ means u has a road to u_i and there is no other RSUs between u and u_i.

we can put the sets of neighbor RSUs to the Cloud Service as a global neighbor RSUs. The global neighbor RSUs

$$H = \{h_i | 0 \leq i \leq n_U\} \tag{3.34}$$

$$h_i \overset{\text{def}}{=} < u_i, H_{u_i} > \tag{3.35}$$

Meet-Table of RSU u is a set of records

$$T_u = \{p_i | 0 \leq i \leq n_{T_u}\} \tag{3.36}$$

$$p_i \overset{\text{def}}{=} < v, t, c, h, c_h > \tag{3.37}$$

Where n_{T_u} is the number of records in T_u. v is a vehicle, $v \in V$, and it passed u c times by time t. $h \in H_u$, v passed h c_h times, and for $\forall h \in H_u$, c_h is the maximum.

In Meet-Fog, global Meet-Table

$$G = \{g_i | 0 \leq i \leq n_G\} \tag{3.38}$$

$$g_i \overset{\text{def}}{=} < v, u > \tag{3.39}$$

n_G is the total number of records in G.

Comparing to Meet-Cloud, records in G is simpler. In fact, records of G in Meet-Fog is generated in Fog, so the Cloud service only need to accept and store records submitted by Fogs. Therefore, the requirements of computing capability, storage, and communication capacity of Cloud Service can be reduced significantly.

3.5.2 Algorithms of Meet-Fog

Based on the formal model of Meet-Fog, algorithms are designed. They are Initiate-System, Manage-RSU, Manage-Vehicle, Distribute-Message, and Redistribute-Message algorithm.

Fig. 19 shows the function of Initiate-System, which is to construct necessary data structures. This function runs only once during the setup process of the system.

All these sets can be stored in the cloud by Cloud Service.

Manage-RSU Algorithm presented in Fig. 20 is used to maintain the set of neighbor RSUs. It is triggered by the added or removed RSU and executed by Cloud Service and all RSUs in a distributed model.

In VANETs, RSUs are a part of the infrastructure, so they often keep static. Since adding and removing RSUs does not often occur, it is applicable to use Cloud Service for managing global neighbor RSUs and handling RSU by adding and removing.

When a vehicle v passes by an RSU u, u will process its Meet-Table T_u. The record in T_u of v contains a neighbor RSU h that v passed mostly. h and other RSUs, v passed, forms a Fog to process the negative message of v. The algorithm Manage-Vehicle shown in Fig. 21 is triggered by a vehicle v approaching RSU u.

When a negative message m needs to be distributed, the Cloud Service invokes the execution of the Distribute-Message algorithm shown in Fig. 22.

Comparing Distribute-Message with Alg-Distribute Algorithm of Meet-Cloud, the Cloud Service only sends m one time in Meet-Fog but many times in Meet-Cloud.

When a vehicle v moves into a new area and encounters a new RSU u, the negative message of v should be redistributed to these vehicles passing u. The redistributing procedure is triggered by the moving vehicle v, and mainly executed by the RSU u. Fig. 23 presents Redistribute-Message Algorithm.

3.5.3 Performance Analysis

Meet-Fog utilizes computing power, communication bandwidth, storing capacity of the edge of VANET to improve the efficiency of negative message distribution. With the basis of formal model and algorithms of Meet-Fog and Meet-Cloud, we can formally analyze message coverage and computing, communication, and storage

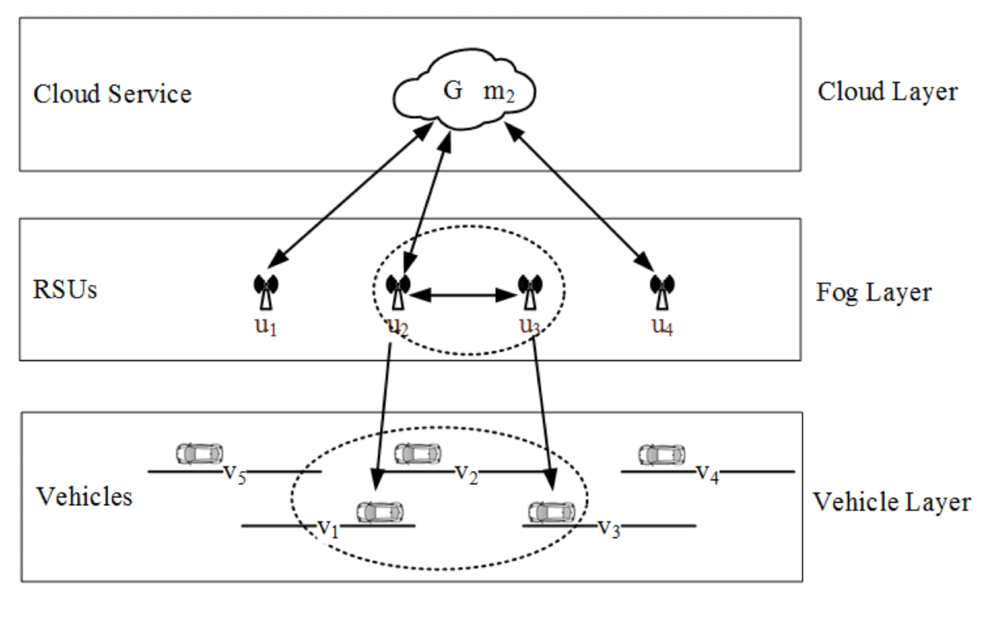
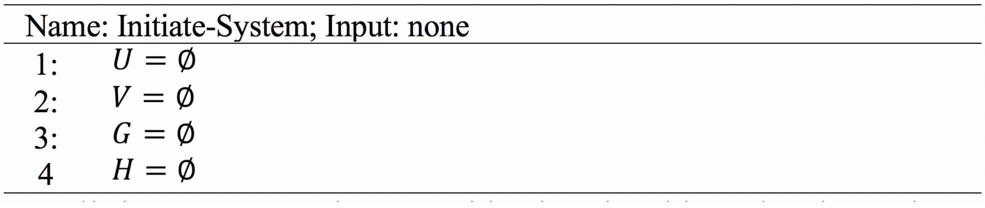
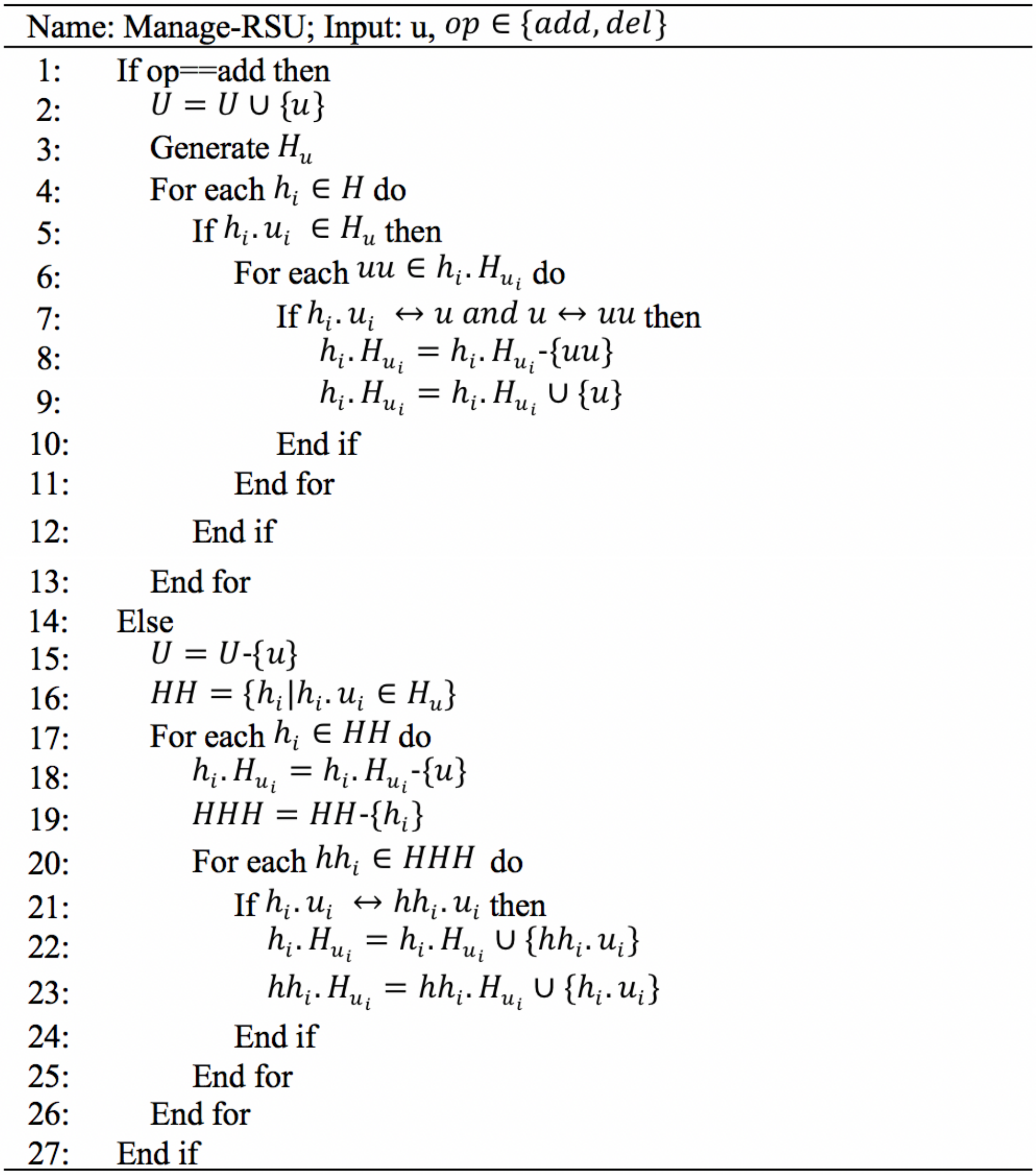


Fig. 18. Architecture of Meet-Fog



```
Name: Initiate-System; Input: none
    $U = \emptyset$
    $V = \emptyset$
    $G = \emptyset$
    $H = \emptyset$
```

Fig. 19. Initiate-System Algorithm



```
Name: Manage-RSU; Input: u, $op \in \{add, del\}$
If op==add then
   $U = U \cup \{u\}$
   Generate $H_u$
   For each $h_i \in H$ do
      If $h_i.u_i \in H_u$ then
         For each $uu \in h_i.H_{u_i}$ do
            If $h_i.u_i \leftrightarrow u \; and \; u \leftrightarrow uu$ then
               $h_i.H_{u_i} = h_i.H_{u_i}$-$\{uu\}$
               $h_i.H_{u_i} = h_i.H_{u_i} \cup \{u\}$
            End if
         End for
      End if
   End for
Else
   $U = U$-$\{u\}$
   $HH = \{h_i | h_i.u_i \in H_u\}$
   For each $h_i \in HH$ do
      $h_i.H_{u_i} = h_i.H_{u_i}$-$\{u\}$
      $HHH = HH$-$\{h_i\}$
      For each $hh_i \in HHH$ do
         If $h_i.u_i \leftrightarrow hh_i.u_i$ then
            $h_i.H_{u_i} = h_i.H_{u_i} \cup \{hh_i.u_i\}$
            $hh_i.H_{u_i} = hh_i.H_{u_i} \cup \{h_i.u_i\}$
         End if
      End for
   End for
End if
```

Fig. 20. Manage-RSU Algorithm

complexity of different negative message distribution schemes. We compare Meet-Fog, Meet-Cloud, Epidemic model[10], and RSU broadcast [59] in this section.

In previous subsection, we created a simulation data set and simulated RSU broadcast, Epidemic, and Meet-Table. The results show that Meet-Table can get high coverage percentage and high accurate coverage percentage at the same time.

Both Meet-Fog and Meet-Cloud are based on Meet-Table. The difference between them is that Meet-Fog moves the task of generating global Meet-Table and distributing the multi copy of negative message among RSUs to the Fog, which consists of RSUs. So, on message coverage, Meet-Fog has the same results of Meet-Cloud. We can put the simulation results in Meet-Cloud and the analysis results together into Table 8. According to Evaluation-Criteria-1 and Table 8, Meet-Fog is a good message distribution scheme.

Table 8. Message Coverage of Distribution Scheme

Value Name	Epidemic Model	RSU Broadcasting	Meet-Cloud	Meet-Fog
Percent of Vehicles Possessing Message	Very high	High	Low	Low
Coverage Percentage	Very high	High	High	High
Accurate Coverage Percentage	Low	Low	High	High

We not only analyze message coverage, but also analyze the algorithm complexity.

For simplicity, we define several average quantities in VANETs as Meet-Cloud. The average number of RSUs a vehicle may encounter and the average number of vehicles an RSU may encounter are same to the definitions in Meet-Cloud part. The average number of neighbor RSUs a RSU may have is

$$\overline{n_h} = \left(\sum_{h_i \in H} |h_i \cdot H_{u_i}| \right) / n_U \tag{3.40}$$

With the help of these average values, we can calculate the complexity of Meet-

Fog. The results are shown in Table 9.

Table 9. Complexity of Distributing Method II

Complexity	Meet-Fog	Meet-Cloud	Epidemic Model	RSU Broadcasting
Core to RSU Communication	n_M	$n_M\overline{n_u}$	N/A	$n_M n_U$
RSU-RSU Communication	$n_m\overline{n_h}$	N/A	N/A	N/A
RSU to Vehicle Communication	$n_m\overline{n_h} \cdot \overline{n_v}$	$n_M\overline{n_u} \cdot \overline{n_v}$	N/A	$n_M n_U\overline{n_v}$
Vehicle to Vehicle Communication	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M n_V^2$	N/A
Core Storage	$n_M + n_v$	$n_M + n_v\overline{n_u}$	N/A	N/A
Edge Storage	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M\overline{n_u} \cdot \overline{n_v}$	$n_M n_V$	$n_M n_V$
Core Computing	N/A	$n_U n_V$	N/A	N/A
Edge Computing	$n_U\overline{n_v}$	N/A	N/A	N/A

From Table 9 we can see that (1) Meet-Fog can sharply reduce the bandwidth requirement of the cloud when it is compared with Meet-Cloud. (2) Meet-Fog can sharply reduce the storage requirement of the cloud when it is compared with Meet-Cloud. (3) Meet-Fog can completely move the computing power requirement of the cloud to the edge when it is compared with Meet-Cloud. (4) Meet-Fog and Meet-Cloud can sharply reduce the communication between vehicles when they are compared with Epidemic, for $\overline{n_u} << n_V$ and $\overline{n_v} << n_V$. (5) Meet-Fog and Meet-Cloud can sharply reduce the storage requirement on vehicles when they are compared with Epidemic and RSU broadcast, for $\overline{n_u} \cdot \overline{n_v} << n_V$ in large scale VANETs.

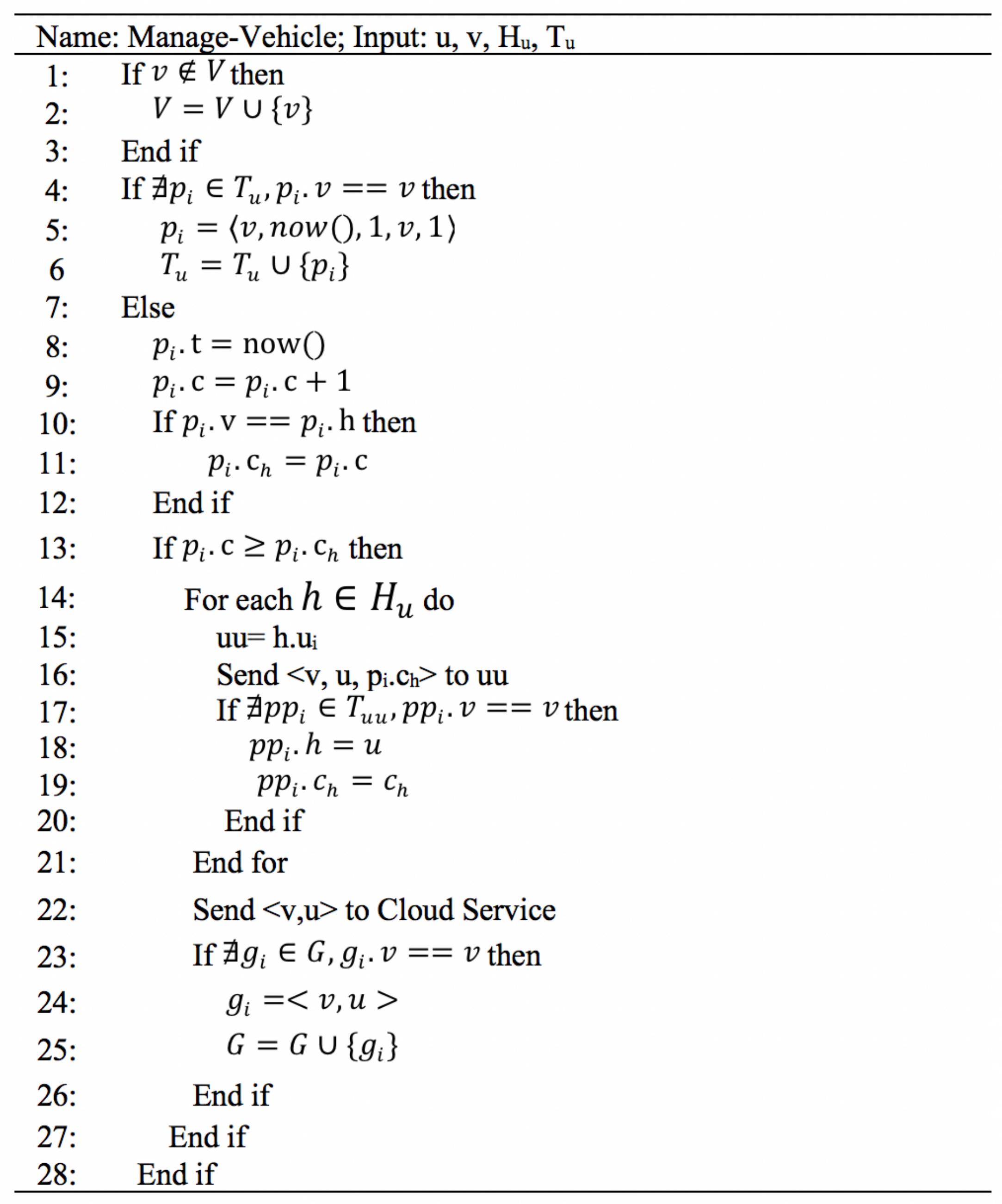
3.6 Conclusion

Accurately and efficiently distributing messages is essential in VANETs. The Epidemic method can rapidly distribute message but may cause flooding storm. The Epidemic with TTL can control flooding storm but may sharply decrease coverage of messages in VANETs. Meet-Table of a vehicle records vehicles it met, meaning the high probability that it will encounter these vehicles. In this section, we utilize Meet-

Table to lead TTL resetting in messages broadcasting. Simulation results show that Meet-Table with TTL for the broadcasting of messages has high coverage percentage, high accurate coverage percentage, and low delay of time. So, Meet-table with TTL can not only increase coverage but also accurately and timely increase coverage.

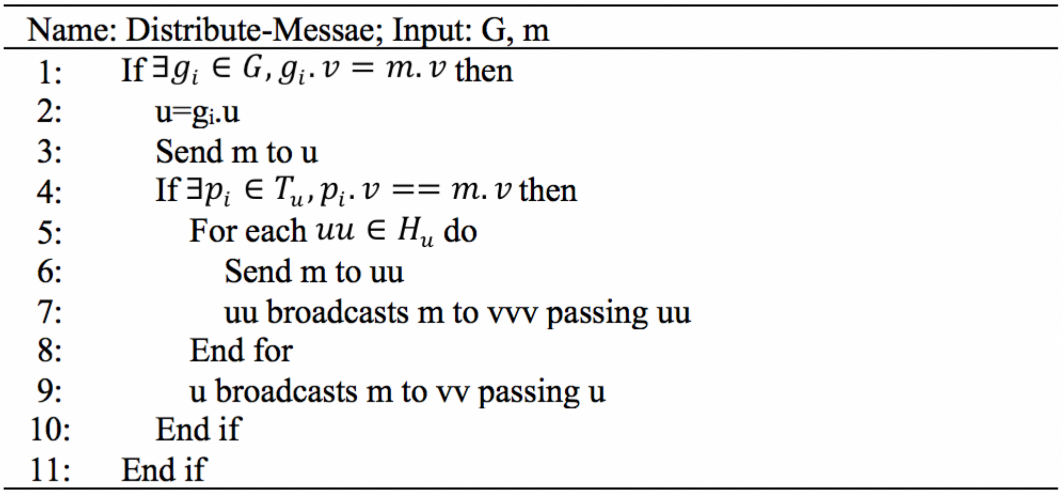
A method combine Meet-Table and Cloud Computing named Meet-Cloud is also proposed. Through formal analysis of proposed Meet-Cloud, we found that Meet-Cloud has low communication and storage complexities than RSU Broadcasting and Epidemic model. Performance simulation results show that Meet-Cloud has both high coverage percentage and high accurate coverage percentage, and mid message delay. The simulation results of fake Meet-Table attack and DoS attack of RSUs show that Meet-Cloud is secure to face these attacks. Therefore, Meet-Cloud is better than RSU Broadcast and Epidemic model in distributing negative messages in VANET. During the analysis of the results of simulation of attacks on the proposed Meet-Cloud, we found that it has attributes of both RSU broadcast and Epidemic. In fact, Meet-Cloud uses the Meet-Table to select several RSUs as the start point of Epidemic.

Then, we propose Meet-Fog, a Meet-Table and Fog Computing based scheme, to utilize Fog Computing for improving our previous Meet-Table and Cloud Computing based scheme. By employing Fog Computing, we move Meet-Cloud to Meet-Fog to make use of resources on the edge of VANET. In Meet-Fog, RSUs are organized as Fog to help Meet-Table management and messages distribution. Meet-Fog advances Meet-Cloud by doing local work locally. As the analysis results show, Meet-Fog is a location aware, distributed, efficient and accurate message distribution scheme for VANETs.



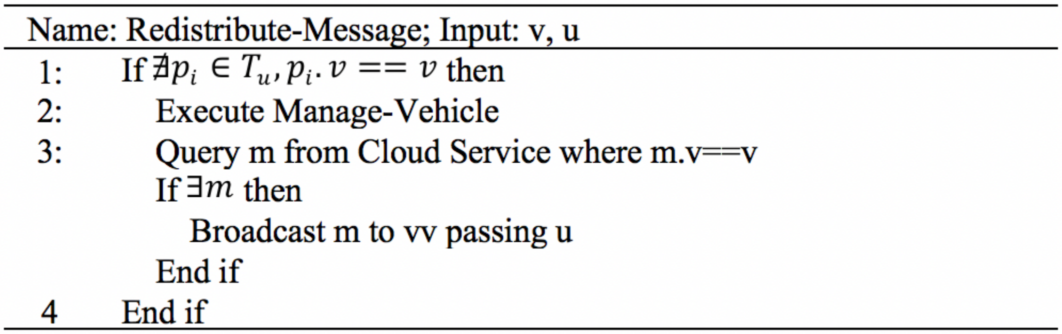
```
Name: Manage-Vehicle; Input: u, v, $H_u$, $T_u$
If $v \notin V$ then
    $V = V \cup \{v\}$
End if
If $\nexists p_i \in T_u, p_i.v == v$ then
    $p_i = \langle v, now(), 1, v, 1 \rangle$
    $T_u = T_u \cup \{p_i\}$
Else
    $p_i.\mathrm{t} = \mathrm{now}()$
    $p_i.\mathrm{c} = p_i.\mathrm{c} + 1$
    If $p_i.\mathrm{v} == p_i.\mathrm{h}$ then
        $p_i.\mathrm{c}_h = p_i.\mathrm{c}$
    End if
    If $p_i.\mathrm{c} \geq p_i.\mathrm{c}_h$ then
        For each $h \in H_u$ do
            uu= $h.u_i$
            Send <v, u, $p_i.c_h$> to uu
            If $\nexists pp_i \in T_{uu}, pp_i.v == v$ then
                $pp_i.h = u$
                $pp_i.c_h = c_h$
            End if
        End for
        Send <v,u> to Cloud Service
        If $\nexists g_i \in G, g_i.v == v$ then
            $g_i =< v, u >$
            $G = G \cup \{g_i\}$
        End if
    End if
End if
```

Fig. 21. Manage-Vehicle Algorithm



```
Name: Distribute-Messae; Input: G, m
If $\exists g_i \in G, g_i.v = m.v$ then
    u=$g_i$.u
    Send m to u
    If $\exists p_i \in T_u, p_i.v == m.v$ then
        For each $uu \in H_u$ do
            Send m to uu
            uu broadcasts m to vvv passing uu
        End for
        u broadcasts m to vv passing u
    End if
End if
```

Fig. 22. Distribute-Message Algorithm



```
Name: Redistribute-Message; Input: v, u
If $\nexists p_i \in T_u, p_i.v == v$ then
    Execute Manage-Vehicle
    Query m from Cloud Service where m.v==v
    If $\exists m$ then
        Broadcast m to vv passing u
    End if
End if
```

Fig. 23. Redistribute-Message Algorithm

CHAPTER 4

TASK I.2: SELF-ORGANIZED CLUSTER-BASED DISSEMINATION

4.1 Problem Statement

Current approaches for vehicle communications are roughly categorized into two classes according to the adopted radio interfaces. One class of approaches is based on Dedicated Short Range Communication (DSRC). The other class is based on existing cellular technology[72].

DSRC began to be used for V2V communication from the 90s. With the rapid improvement of mobile cellular networks, cellular technologies also catch some researchers' attention to use the existing cellular infrastructures and technologies for vehicle communication.

Unfortunately, both DSRC and mobile cellular networks cannot fully meet the needs of ITS. DSRC has a shortage in medium range. It is inadequate for large-scale deployment[73] because of its coverage radius is not large enough. Mobile cellular networks provide wide and large coverage, while its delay is longer than DSRC for real-time information exchanges in local areas[74].

As a result, DSRC and mobile cellular networks are combined for vehicular network communications. Fig. 24 shows an example of vehicular network, which supports communication not only via LTE but also via DSRC [74]. Overall, vehicular networks are centralized as clusters because of cellular is based on connections and scheduling. Vehicles may also exchange messages with their neighbors via DSRC. As a result, vehicular networks still have decentralized parts under the centralized architecture.

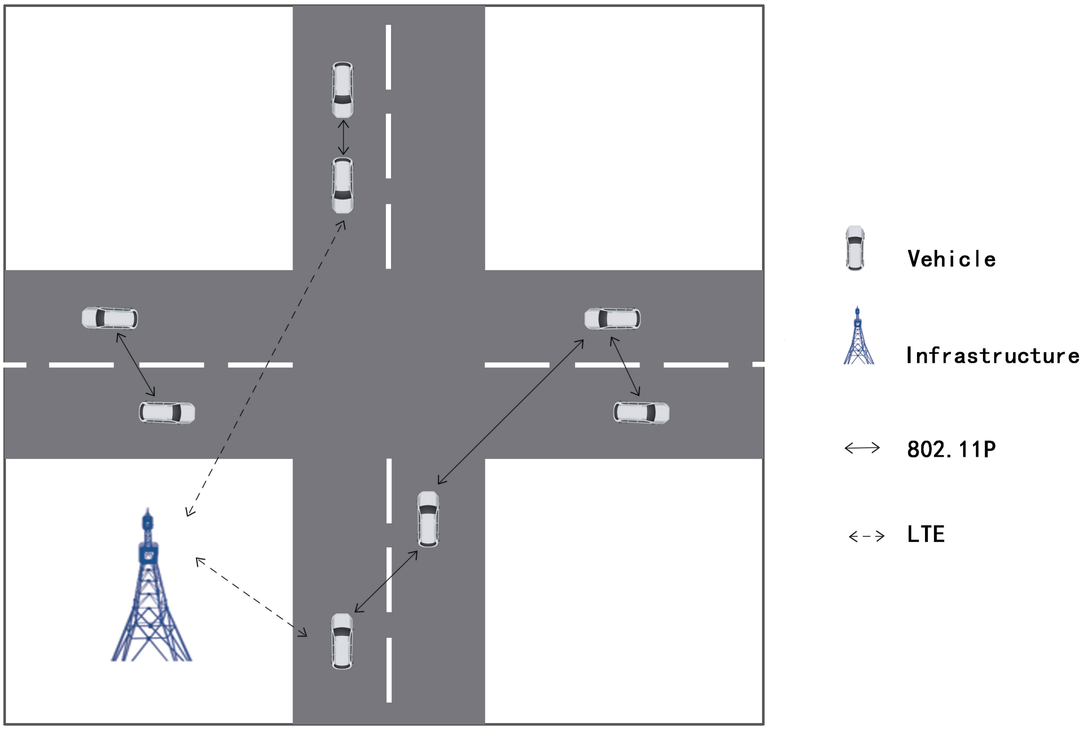


Fig. 24. An Example of Vehicular Network

Many frameworks for managing vehicular networks have been proposed. Dividing vehicles into clusters is a common and reasonable approach. [19] proposes a novel framework named LTE4V2X. All vehicles have two interfaces. One is LTE interface. The other is 802.11p interface. Vehicles are signed into clusters. The size of cluster is smaller than or equal to the range of 802.11p. So that, vehicles in the same cluster can exchange messages via DSRC. DSRC coverage radius is about 300 meters. LTE coverage radius is about 1 kilometers. Therefore, a single eNodeB manages many clusters around it. Within a cluster, a vehicle performs as a Cluster Head (CH) to collect information of all Cluster Members (CM) via 802.11p and exchanges data with the eNodeB via TLE. Fig. 25 is a simplified view of a cluster-based vehicular network.

This framework creates much less overhead and increases efficiency. [19] selects

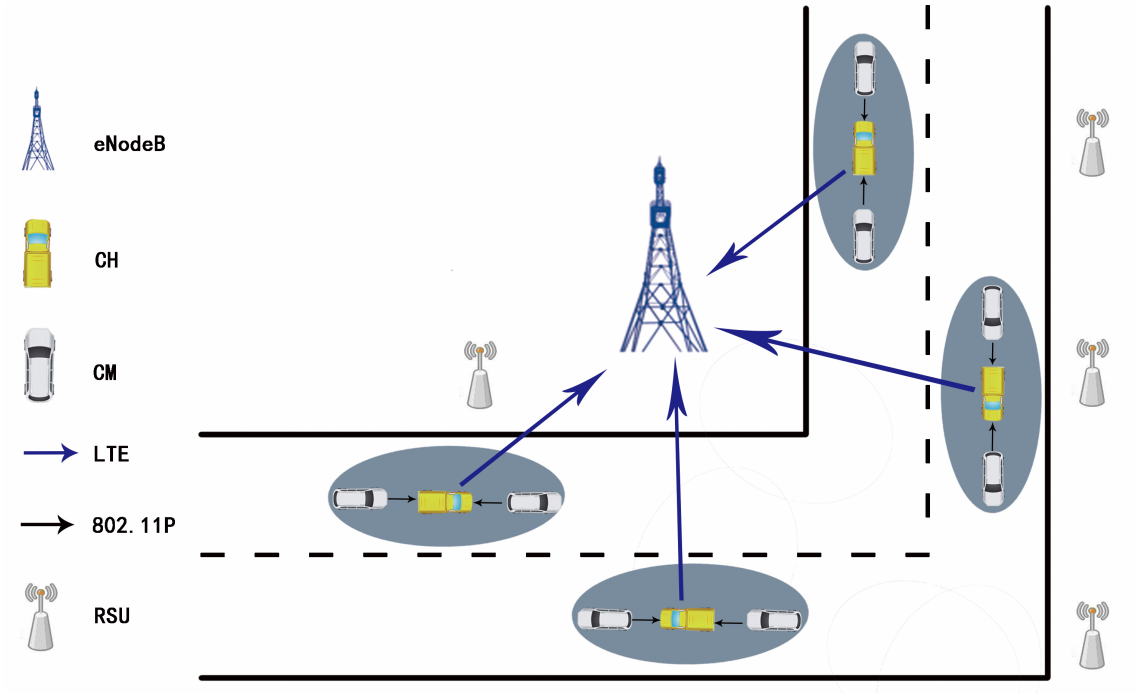


Fig. 25. A Simplified View of Clustered Vehicular Networks

the node with shorter communication delay to be CH. To find this node, vehicles broadcast a CH_ANNOUNCE message. If a vehicle hears CH_ANNOUNCE message, it will not broadcast its CH_ANNOUNCE message. The vehicle who first broadcasts a CH_ANNOUNCE message will be selected as CH. Although this method reduces communication delay in vehicular networks, it has the disadvantages such as infrastructure cannot know which one is CH until CH sends a message to it; information collision may happen because all vehicles may broadcast their announcements; the lifetime of a cluster is not very long. Therefore, we try to improve the clusters forming and Cluster Head selection method.

In cluster-based vehicular networks, all vehicles send their position information to eNodeB. Then, eNodeB manages the vehicles by clusters. CH performs as a messenger to help eNodeB and CMs exchange information. We assume all vehicles are able to communicate via both LTE and DSRC. Fig. 26 illustrates the communications within one cluster. First, CH receives a request from eNodeB via LTE. Second, CH broadcasts the request via 802.11p. Then, CMs unicast the information to CH via 802.11p. After collecting the information from all CMs and compressing the information, CH sends the information back to eNodeB via LTE again.

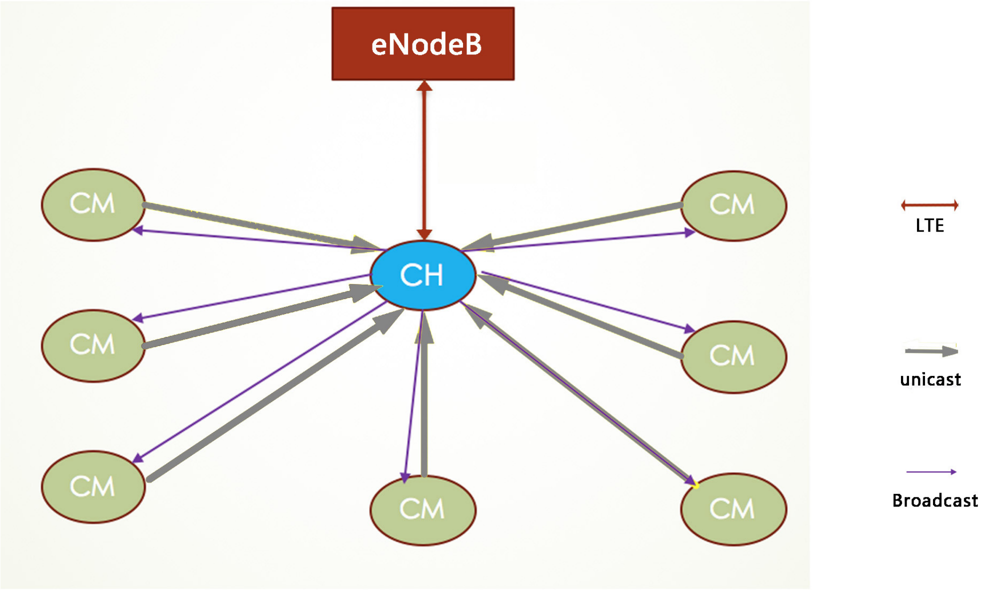


Fig. 26. Communications within One Cluster

Compared with other MANETs, nodes in VANETs have higher mobility and higher speed. Cluster reforming and CH changing must be much more frequently than other typical MANETs. To decrease the management overhead and in- crease communication quality, the clustering algorithm for VANETs should be able to form stable clusters. To achieve this goal, this paper proposes a stable clustering algorithm for VANETs. In this paper, we proposed a novel approach to form and maintain stable clusters for VANETs on highways to avoid continual cluster reforming. A center-based clustering algorithm is used to locate the initial clusters' centers. In every cluster, a suitable CH is chosen by vehicles' position, speed, and acceleration. A cluster maintenance algorithm is proposed to keep CMs in its CH's transmission range.

4.2 Proposed Approach

In vehicular networks on highways, vehicles have very high mobility while the road environment is simpler than other areas. We use a center-based clustering algorithm to locate the initial cluster centers which is close to the densest areas. In every cluster, a suitable CH is chosen with the help of vehicles' position, speed, and acceleration. A cluster maintenance algorithm is also proposed to avoid continual cluster reforming.

4.2.1 Overview and Assumption

Clustering algorithm groups a set of unlabeled nodes into clusters. In cluster-based VANETs, all vehicles send their position information to eNodeB. Then, eNodeB manages the vehicles by clusters. CH performs as a messenger to help eNodeB and CMs exchange information.

In this task, we propose a center detection based clustering algorithm. We group the vehicles in the region where the density of vehicles is higher than other areas into clusters with the help of blob detection method or an improved high-degree algorithm. Some parameters, such as speed and acceleration, are added to the CH selection metric to make the cluster stabler and decrease the CH re-selection frequency.

In this task, we have some assumption:

1. All vehicles have both LTE and 802.11p interfaces.

2. All vehicles are equipped with Global Positioning System (GPS) devises. So, they have accurate geolocations.

3. All vehicles know their destination, speed, and acceleration. n j

4.2.2 Cluster Formation

In our proposed algorithm, in the initialization stage of cluster formation, vehicles send beacon messages to the eNodeB. The beacon message of one vehicle contains the vehicle's ID k, current position (x_k, y_k) , current speed v_k, maximal acceleration a_k and direction type t_k.

Direction type is decided by the vector from the current position to the destination. For vehicle k, whose destination position is (x'_k, y'_k), the direction vector

$$\overrightarrow{v}_k = (x'_k - x_k, y'_k - y_k) \tag{4.1}$$

The Direction Angle θ_k of vector $\overrightarrow{v}_k$ is

$$\theta_k = tan^{-1} \frac{y'_k - y_k}{x'_k - x_k} \tag{4.2}$$

When $\theta_k \in [0°, 90°)$, $t_k = 1$. When $\theta_k \in [90°, 180°)$, $t_k = 2$. When $\theta_k \in [180°, 270°)$, $t_k = 3$. When $\theta_k \in [270°, 360°)$, $t_k = 4$. Vehicles have different t are managed respectively.

The initial clustering algorithm is described in Algorithm Fig. 27.

After receiving the beacon messages, the system analyzes vehicles' position information, and detect the centers of the regions where the vehicle density is higher than other areas. If the vehicle quantity or the vehicle density are not very large, an improved Highest-Degree Algorithm is applied. Several vehicles which have more neighbors in their transmit range are detected. We improve the original Highest-Degree Algorithm to make sure the distance between any two vehicles we detected is larger than the DRSC range. The positions of detected vehicles will be the centers we use in the clustering algorithm. Otherwise, when the vehicle quantity and the vehicle density are very large, to decrease the computing complexity and analyze time, the system draws dots on the map to indicate vehicles. The blob detection algorithm[75] is used to detect the centers of regions on the map where the gray pixel value is greater.

All vehicles whose distance to the center are not larger than the range of DSRC are labeled as one cluster. Then, the system selects one nearest intersection for every center among all intersections meet the following conditions:

1. The distance from it to the points in P is not smaller than the range of DSRC.

2. The intersection is not in any cluster's region.

Vehicles near those selected intersections are grouped into clusters. Then, eNodeB uses the same way to select intersections near the selected intersections and groups vehicles. After iterations, ungrouped vehicles are grouped into clusters. The distance between two vehicles in the same cluster is not larger than the range of DRSC. To further decrease computing complexity, in line 9 of Clustering Algorithm, a vehicle or infrastructure located in the center or intersection can broadcast a request to invite neighbors to join the cluster. In line 28, the chosen vehicle e can broadcast an invitation instead of calculating distance by the system.

4.2.2.1 Cluster Head Selection

Compare to other MANETs, VANETs have lower stability, because of the high mobility of vehicles. Although we divide the vehicles with the help of direction vector $\overrightarrow{v}_k$, the stability of clusters cannot be guaranteed. To select an appropriate CH which can increase the cluster lifetime and decrease the CH reselecting frequency, a relative mobility metric $\mathbf{M}$ is introduced for CH election.

For a vehicle k, that is in the cluster $cluster_i$, the position differences between it to all other N vehicles in the same cluster $cluster_i$ is

$$D_k = \sum_{n=1}^{N} \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2} \tag{4.3}$$

The speed differences between k to all other N vehicles in the same cluster is

$$V_k = \sum_{n=1}^{N} |v_k - v_n| \tag{4.4}$$

The maximal acceleration differences between k to all other N vehicles in the same cluster is

$$A_k = \sum_{n=1}^{N} |a_k - a_n| \tag{4.5}$$

The relative mobility metric $\mathbf{M}$ is

$$\mathbf{M}_k = \alpha \frac{D_k}{max\left\{D_n | \forall n \in C_i\right\}} + \beta \frac{V_k}{max\left\{V_n | \forall n \in C_i\right\}} + \gamma \frac{A_k}{max\left\{A_n | \forall n \in C_i\right\}} \tag{4.6}$$

, where α, β, and γ are the weighted coefficients. $\alpha + \beta + \gamma = 1$. They can be adjusted to fit the different traffic conditions.

The relative mobility metric $\mathbf{M}$ evaluates the relative position, speed and maximal acceleration differences between one vehicle to all other vehicles in the same cluster. A smaller $\mathbf{M}$ indicates the vehicle has lower relative mobility than other vehicles in this cluster. Cluster Head Selection Algorithm shown in Fig. 28 explains the process of Cluster Head selection. All clusters formed with the help of centers and intersections use Cluster Head Selection Algorithm to select CH. As a CH, the vehicle's relative mobility metric is smaller than any CMs. That means the motion mode of CH is similar to the whole cluster.

4.2.2.2 Cluster Maintenance and Reforming

The unpredictability and mobility of traffic make the cluster lifetime temporary. It is infeasible to reform clusters in real time or very frequently. To minimize the frequency and overhead of cluster reforming, we propose a cluster maintenance algorithm. Cluster Maintain Algorithm shown in Fig. 29 explains the cluster maintenance process.

1)No connections between CH and CM

When a CH cannot connect to a CM, the CH will delete the CM from its record and notice eNodeB. When a CM cannot reach its CH, the CM will check the signal it received via DSRC, and join the cluster whose signal of CH is strongest. If the CM cannot receive a message strong enough, it will notice eNodeB via LTE and become a CH.

2)No connections between eNodeB and CH

When eNodeB notices it has lost connection to a CH, it recalls Cluster Head Selection Algorithm and a new vehicle will be CH of that cluster instead of the leaving vehicle.

3)A vehicle joins the network

When a vehicle comes into the network, it first tries to join the nearest cluster by broadcasting a CH request via DSRC. If it fails, it will send a message to eNodeB. eNodeB will help the vehicle to join a cluster, or to be a CH and form a new cluster by itself.

4)Two clusters are too close

With the movement of the vehicles, two clusters may be very close. When the distance between two CHs is shorter than R for a period Δt, the two clusters are merged into one cluster. The Cluster Head Selected Algorithm is recalled. A new CH for the new cluster is selected. Then, all vehicles, which are out of the transmission range of the new CH, leave this cluster and check the invitation signal they have received via DSRC, and join the cluster whose signal of CH is the strongest. If a vehicle does not find a cluster to join in, it notices eNodeB via LTE and becomes a CH.

4.3 Performance Evaluation

4.3.1 Simulation Parameter

We perform the simulation with the help of Veins LTE. Veins LTE is a simulator developed on Veins [76], which is an open source framework for simulation of vehicular networks based on both IEEE 802.11p and LTE. It integrates a network simulator named OMNeT++ and a traffic simulator named Simulation of Urban MObility (SUMO) [77].

In our experiment, vehicles run on a real map of Washington, D.C., USA, obtained from OpenStreetMap[78]. We extract the data of highways in the center of Washington, D.C.. The total length of road is 30.38 km. The total lane length is 90.09 km. Every vehicle has random source and destination edge. The route from the starting point to the destination is the shortest path found by Dijkstra's algorithm[79]. The maximal acceleration ability of vehicles we have used is 2.6 m/s^2. The maximal deceleration ability of vehicles is 4.5 m/s^2. The vehicle's maximum velocity is 55.55 m/s.

We compare our proposed clustering algorithm, Center-Based Stable Clustering Algorithm (CBSC), with a K-Means-Based method (KMB) and SCalE algorithm[61]. K-means algorithm[80] is commonly used in VANETs for clustering, e.g. [81], [82], and [83]. In KMB method, we divide the vehicles into two parts by the angle of the vehicles and perform KMB on them respectively. The cluster maintenance algorithm KMB uses is proposed in [62]. The predefined threshold Δv_{th} is 5 m/s. In the simulations, all vehicles' movement information is resent to eNodeB for cluster status update in every 10 seconds. eNodeB needs exchange data with vesicles in every 3 seconds. The simulation time is 503 seconds.

4.3.2 Results and Analysis

The goal of this paper is to propose a stable clustering algorithm for VANETs. To check whether a clustering algorithm can solve the high mobility of vehicles on the highways, the cluster stability should be evaluated. The metrics we use to show the performance of clustering algorithm are as follows:

1) Average CH Lifetime: The CH lifetime is the period from the vehicle to be a CH to it is not a CH (i.e., be a CM or leave the system). When a CH ends its lifetime, a new CH is elected, or the cluster is dissolved.

2) Average CM Lifetime: CM lifetime represents the duration of a CM stays in the same cluster. The average CM lifetime is the average length of all vehicles' CM lifetime. It is another important metric to evaluate the stability of clusters.

3) Average Number of Re-affiliation Times per Vehicle: The average number of re-affiliation times per vehicle represents the average number of times a vehicle changes the cluster it belongs to during the simulation time.

4) Packet Loss Rate: Packet loss rate is the percentage of packets lost with respect to packets sent.

In the experimentation, we compare the four metrics of the three methods with different vehicle numbers, transmission ranges, or highway speed limits. Fig. 30, Fig. 32, Fig. 34, and Fig. 36 show the results obtained with the variety of total vehicle number (N) and the variety of transmission range (R), when the highway speed limit

(v) is 100 km/h. Fig. 31, Fig. 33, Fig. 35, and Fig. 37 show the results obtained with the variety of transmission range (R) and the variety of highway speed limit (v), when the total vehicle number (N) is 300.

Fig. 30 and Fig. 31 represent the average CH lifetime for the three methods. Those figures enlighten the CHs under KMB have a marked shorter lifetime. Although our CBSC has a higher value than SCalE a few times, in general, SCalE performs slightly better than CBSC on the average CH lifetime.

The average CM lifetime values produced by KMB, SCalE and the CBSC methods are shown in Fig. 32 and Fig. 33. From those two figures, we can see that the average CM lifetime produced by CBSC is much longer than other two methods. ScalE has the worst performance on the average CM lifetime.

Fig. 34 and Fig. 35 show the average number of re-affiliation times per vehicle obtained in 503 seconds. Obviously, comparing to other two algorithms, vehicles with ScalE change status much more frequently. The data on the two figures shows CBSC not only produces a lower cluster status change frequency than KMB produces, its superiority but also is bigger with the increase in highway speed limit.

The results of simulation illustrate that clusters under CBSC are the stablest in the three algorithms. They have the longest average CM lifetime and lowest average number of re-affiliation times per vehicle. Although SCalE performs slightly better than CBSC on the CH lifetime experiment, it produces a much shorter average CM lifetime. Besides, the number of CMs is much larger than the CHs in one system. Therefore, we consider that CBSC has higher stability than SCalE.

The basic function of VANETs is allowing communication between separated vehicles and infrastructures. To test the performance of data dissemination in VANETs, we do experiment on packet loss rate with different methods. Packet loss means a packet fails to arrive at its destination. A high packet loss rate decreases the data dissemination efficiency and may cause network congestion. Therefore, an efficient data dissemination mechanism should have a low packet loss rate. In our experiment, all vehicles exchange data with eNodeB every three seconds. That means in every three seconds, eNodeB sends data to all vehicles once, and each vehicle sends data to eNodeB once. Like the scene we described in the previous section, eNodeB communicates with the nodes in its record via CHs, and vehicles which are CMs send data to their CHs first. Fig. 36 and Fig. 37 show the results of packet loss rate. With the increase in vehicle velocity or the transmission range, the packet loss rates obtained by all the three mechanisms decrease. But CBSC gets lower packet loss rate, while KMB performs the worst, when the amount of vehicle is larger. That insinuates CBSC has a good ability to handle a considerable amount of data. In the experiment, CBSC always obtains lowest packet loss rate. Since the interval between cluster status updates is 10 seconds, we can know that the probability of CM leaving its CH between two cluster status updates in CBSC is lower than other two algorithms. We can consider that the proposed relative mobility metric $\mathbf{M}$ and CH selection algorithm of CBSC do reduce the impact of vehicle mobility to cluster stability.

4.4 Conclusion

To decrease the management overhead and increase the quality of communications, we try to make the clusters in VANETs as stable as possible while keeping the network performance acceptable. In this task, we propose a stable clustering algorithm for VANETs on highways, which utilizes direction vector, the centers of vehicle denser areas and intersections to group less quantity of more stable clusters. To reduce the impact of vehicle type and drivers' driving habits, we propose a novel CH selection algorithm and cluster maintenance algorithm, which use the relative mobility metric to reduce the influence of vehicle's distance, velocity, and maximal acceleration. In the simulation experiment, our algorithm's performance ranks up against other two algorithms' (KMB and SCalE) on both stability and package delivery rate. In the future, we would like to further improve the algorithm for the complex urban environment.

```
Algorithm 1: Clustering Algorithm
  Input: Vehicle set V
  Output: Initial clusters
  Initialize center set $C = \phi$;
  Locate the centers;
  Add the centers into $C$;
  Initialize point set $P = C$;
  while $P \neq \phi$ do
      foreach point p in P do
          Initialize node set $cluster_p = \phi$;
          foreach vehicle e in V do
              if $d_{ep} \leqslant R$ then
                  Add $e$ into $cluster_p$;
                  Remove $e$ from $V$;
              end
          end
          if $cluster_p \neq \phi$ then
              Call Algorithm 2;
              Return set $cluster_p$;
              Add the intersection nearest to $p$ which meets
                the conditions into $P$;
          end
          Remove $p$ from $P$;
      end
  end
  while $V \neq \phi$ do
      foreach point c in C do
          Select an element $e$ in $V$ nearest to $c$;
          Initialize set $cluster_e = \{e\}$;
          Remove $e$ from $V$;
          Set $e$ as CH;
          foreach vehicle v in V do
              if $d_{ev} \leqslant R$ then
                  Add $v$ into $cluster_e$;
                  Remove $v$ from $V$;
              end
          end
          Return $cluster_e$;
      end
  end
```

Fig. 27. Clustering Algorithm

```
Algorithm 2: Cluster Head Selection Algorithm
Input: Vehicles in one cluster
Output: Cluster head o of the corresponding cluster
Set $\mathbf{M}_{min} = +\infty$;
foreach vehicle $k$ do
    Calculate the relative mobility metric $\mathbf{M}_k$;
    if $M_k < M_{min}$ then
        $\mathbf{M}_{min} = \mathbf{M}_k$;
        $o = k$;
    end
end
return o;
```

Fig. 28. Cluster Head Selection Algorithm

```
Algorithm 3: Cluster Maintenance Algorithm
if the eNodeB can not reach a CH then
    Call Cluster Head Selection Algorithm;
end
if the CH can not reach a CM then
    Reomve the CM;
    Notice eNodeB;
end
if the distance between two CHs $\leqslant R$ for a period $\Delta t$ then
    Merge the two clusters into one cluster;
    Call the Cluster Head Selected Algorithm;
end
if a CM can not reach the CH then
    if it can receive a signal from CHs then
        Join the cluster whose signal of CH is strongest;
    end
    else
        Notice eNodeB;
        The node performs as a CH;
    end
end
```

Fig. 29. Cluster Maintenance Algorithm

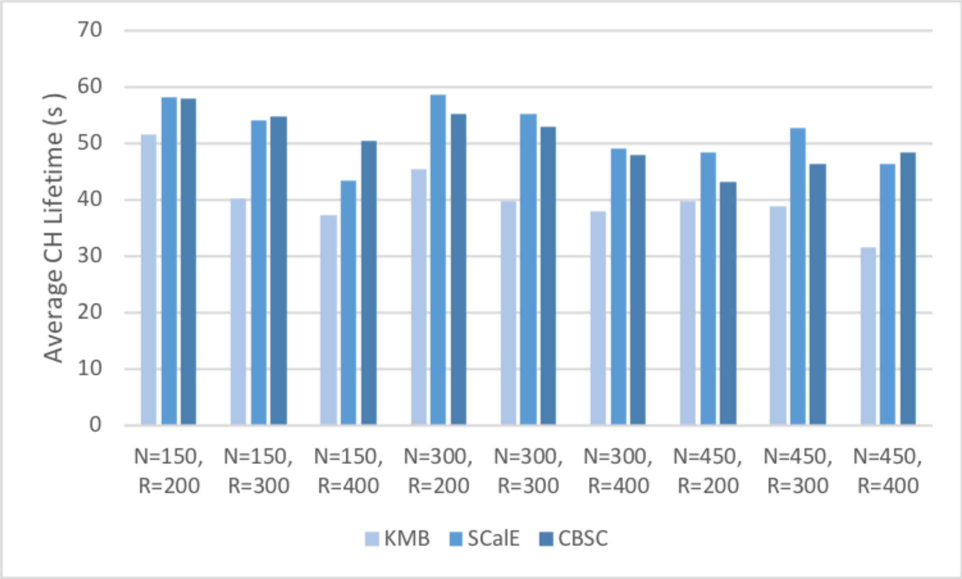


Fig. 30. Average CH Lifetime VS. N and R

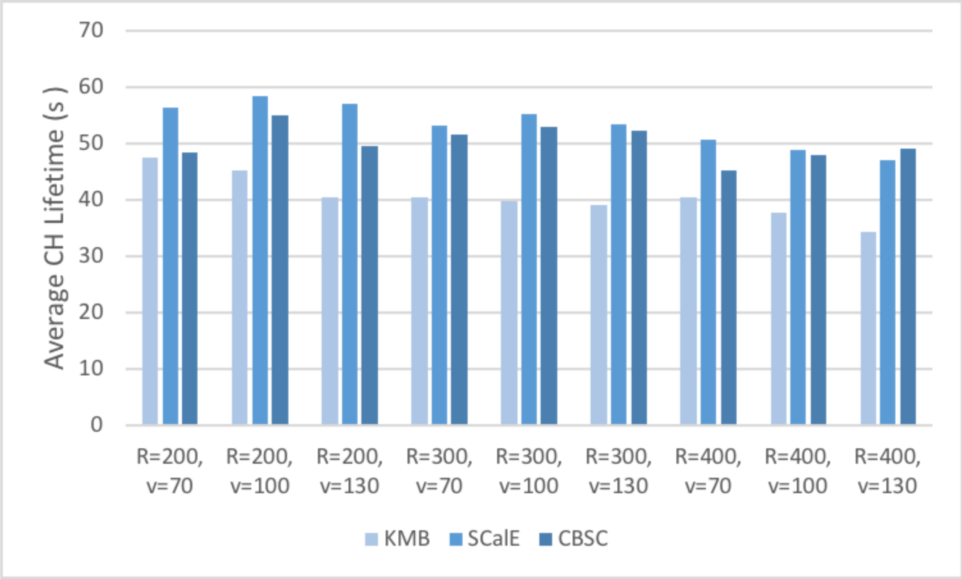


Fig. 31. Average CH Lifetime VS. R and v

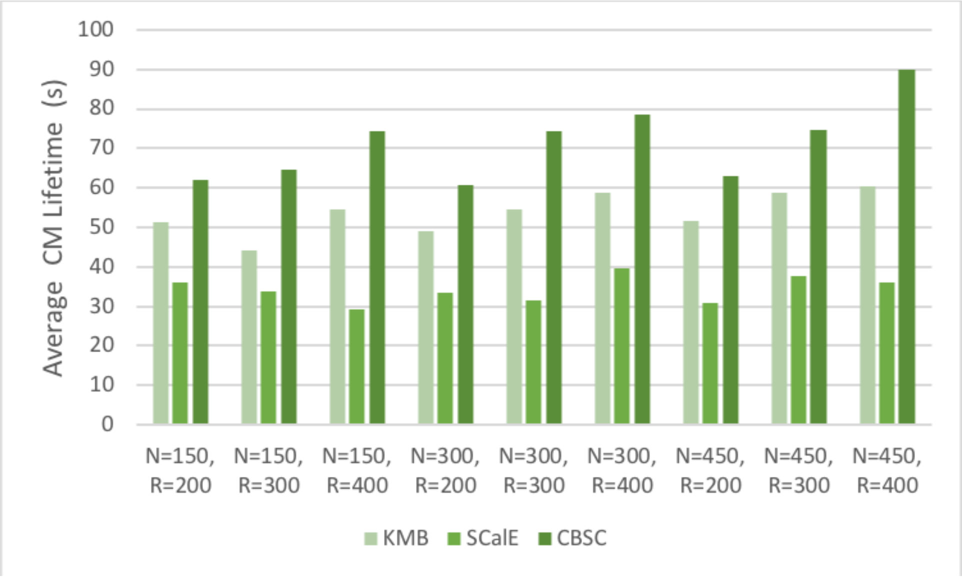


Fig. 32. Average CM Lifetime VS. N and R

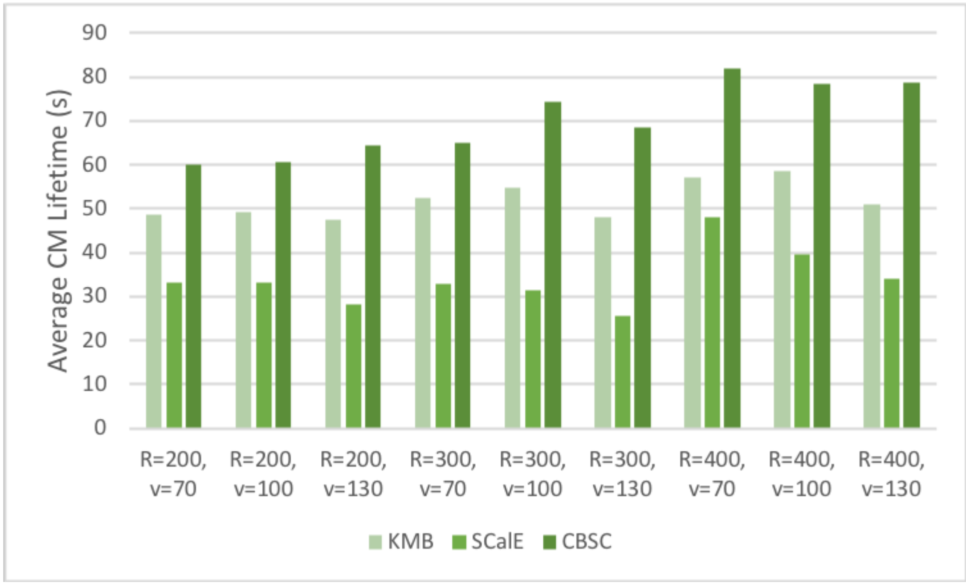


Fig. 33. Average CH Lifetime VS. R and v

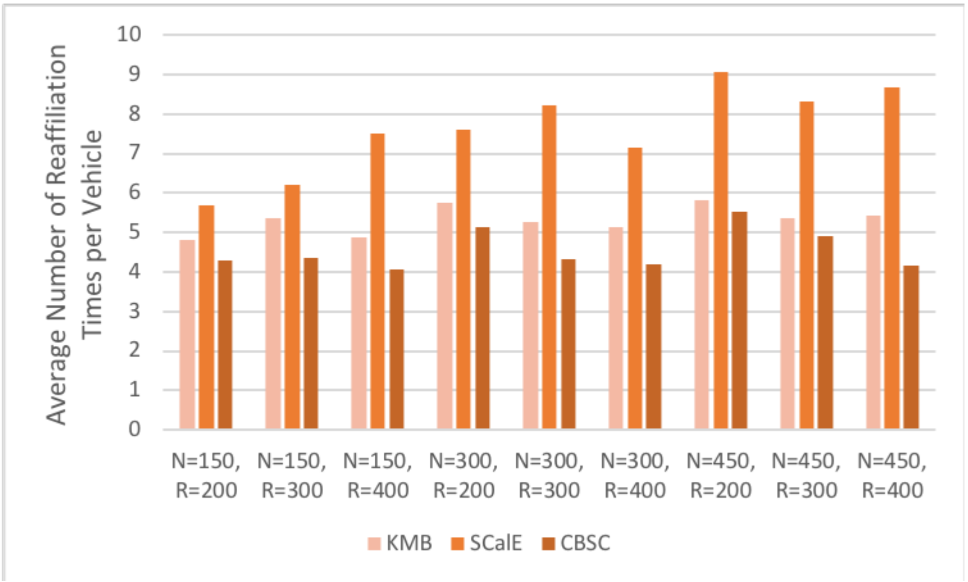


Fig. 34. Average Number of Re-affiliation Times per Vehicle VS. N and R

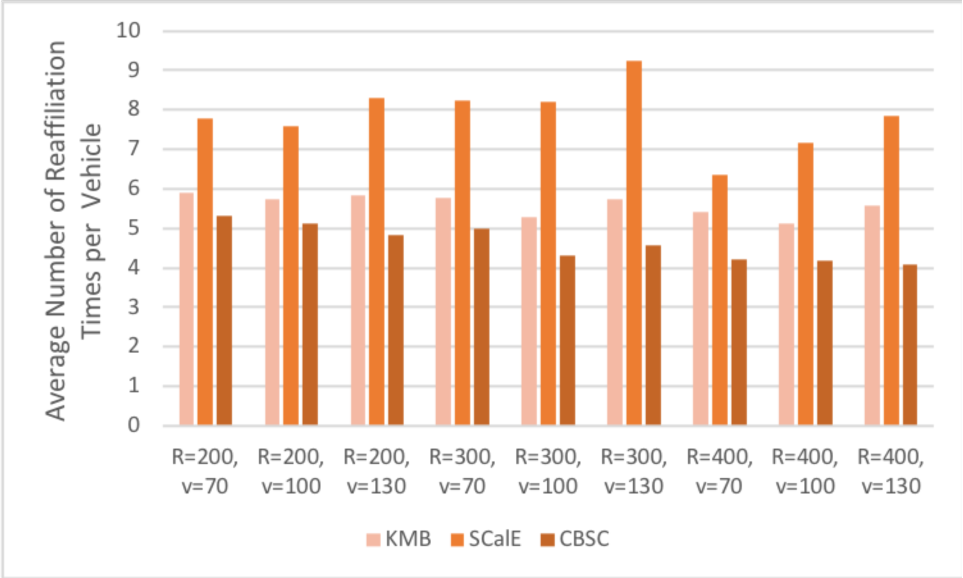


Fig. 35. Average Number of Re-affiliation Times per Vehicle VS. R and v

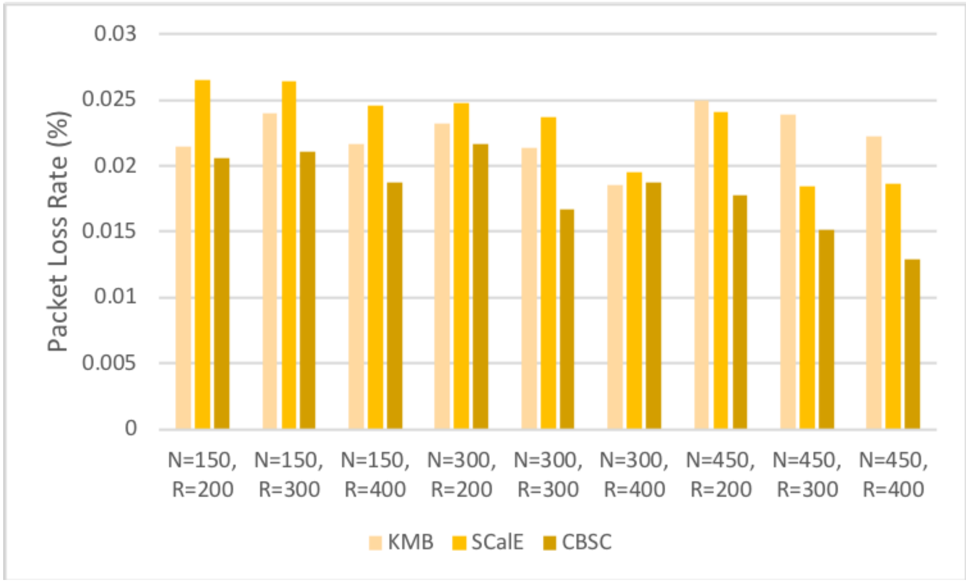


Fig. 36. Packet Loss Rate VS. N and R

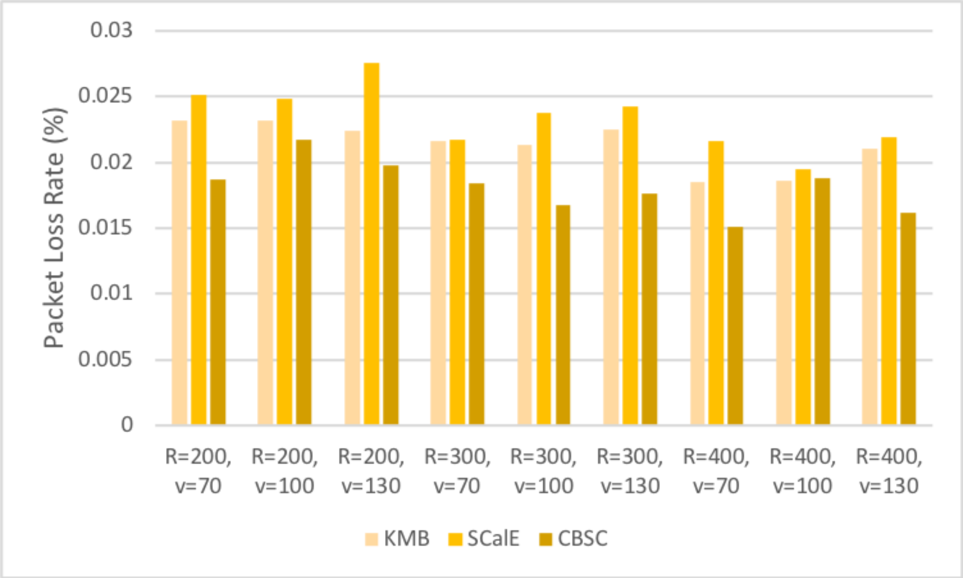


Fig. 37. Packet Loss Rate VS. R and v

Thrust II: Security and Privacy Protection in VANETs

CHAPTER 5

TASK II.1: IMPROVING SECURITY AND STABILITY OF AODV WITH FUZZY NEURAL NETWORK IN VANETS

5.1 Problem Statement

The Ad hoc On-Demand Distance Vector (AODV) is a routing protocol for the nodes in MANETs. It was proposed in 2003 by Perkins et al. [84] provides some details and analyses for a MANET. The MANET has tens to thousands of nodes, and all nodes are trustable.

There are four message types in AODV. They are Route Requests (RREQs), Route Replies (RREPs), Route Error (RERRs), and Route Reply Acknowledgment (RREP-ACK). An RREQ message contains the information about the request ID, transmitting methods, the hop number from the originator to the node handling the request, IP address of destination and originator, and the sequence numbers of destination and originator. An RREP message contains the information about transmitting methods, subnet prefix, hop number, IP address of destination and originator, the sequence numbers of destination and originator, and the valid route lifetime. An RERR message contains the No delete flag, the unreachable destinations number, unreachable destination IP address, and unreachable destination sequence number. If an RREP message requires acknowledgment, an RREP-ACK Message will be sent back to notice that the links may be unidirectional.

In the networks, each node uses route table to record route information. A node creates a new route table entries after creating or getting a route to a new destination. When it receives a fresher route or shorter route, the route table should update. When nodes receive messages, they can use the sequence number to check whether the information is fresher than the previous one. Therefore, route table entries in each node also record the latest sequence number of the corresponding destination. When the node gets a new sequence number from RREQ, RREP or RERR messages, the route table entry will update. The nodes in the networks must maintain its destination sequence number. To make sure the route is valid when it is used, the routing table entry also contains its lifetime and a list of precursors.

When a node wants to communicate with another node which it can not find a valid route to, the node will create and broadcast an RREQ message. Then, the node waits for an RREP with the route to the destination. If the waiting time is longer than NET_TRAVERSAL_TIME, it will rebroadcast a new RREQ message. Then, after 2*NET_TRAVERSAL_TIME, it resends another RREQ message. And so on. When the number of resending RREQ messages is higher than RREQ_RETRIES, the node will consider the destination to be unreachable.

When a node received an RREQ message, it checks its ID and originator IP address. If the originator is itself or it has already received one same request, the node discards this message. If not, the node searches, creates or updates the reverse route from it to the originator. If the node has a valid route or it is the destination, it will create and transmit an RREP message to the next hop to the originator. Sometimes, an intermediate node should send a gratuitous RREP message to the destination after sending an RREP message to the originator. The destination and originator of the gratuitous RREP are the originator and destination of original RREQ message.

After receiving an RREP message, the node finds or creates a route to the previous hop. If this node does not have a route to the destination, the node will create one. Otherwise, it will find the fresher route between the route it records and the route stored in the RREP message, and use the fresher one to replace another. If required, the node also needs to send an RREP-ACK message back when the link to the originator has a possibility to be unidirectional or cause an error. If an RREP message transmission failure occurs, the node will add the next-hop into its blacklist for a period. RREQ messages send by the nodes in blacklist are ignored.

A node, which is a part of an active route, periodically broadcasts the HELLO messages to its neighbors. A HELLO message is a kind of RREP. The destination of HELLO message is the node itself. Hop Count is 0. Its TTL is just 1. If a node does not get a HELLO message from one neighbor for a period, the node considers that the connection to that neighbor is broken. If a node receives a HELLO message from a new node, it can create a new route. If a node receives a HELLO message from a recorded neighbor, it can update the route lifetime and Destination Sequence Number.

A node in an active route also has the duty to keep a close watch on the next hops. When it detects that the link to its next hop is broken, the next hop is considered to be lost. Then, the node sends an RERR message to its precursors. AODV also provides detailed methods to handle some other problems and repair the broken links.

AODV is one of the most common routing protocol for mobile ad hoc networks. It makes sure the route is fresh and overhead is low. However, it only suits for small ad hoc networks. AODV cannot handle MANETs has a large number of nodes with high mobility, such as VANETs. Literature [85] conducts a simulation to evaluate its performance in real urban environment. The results illustrate some improvement is necessary for applying AODV in VANETs.

5.2 Proposed Approach

5.2.1 Improving AODV Protocol with Fuzzy Neural Network

5.2.1.1 Arithmetic Statement

Stability and security judgment throughout the improved AODV. Its success is directly related to the smooth operation of the entire routing protocols. In this task, GASA-FNN is used to improve the scheme based on Genetic Simulated Annealing Algorithm (GASA). Firstly, factors impact security and stability of nodes are taken as the node's security measures and stability measures, and they are normalized. Then, fuzzy neural networks are used to carry out fuzzy calculations on safety metrics and stability metrics, and genetic simulated annealing is used to optimize parameters used in the calculation process. Finally, node stability and node trust values are obtained and discriminated in the routing process. The system structure is shown in Fig. 38.

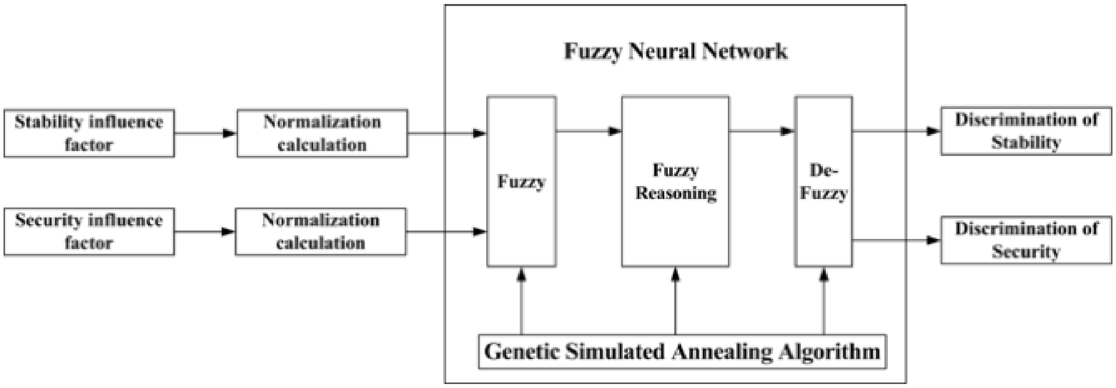

Fuzzy Neural Network
Stability influence factor
Normalization calculation
Security influence factor
Normalization calculation
Fuzzy
Fuzzy Reasoning
De-Fuzzy
Discrimination of Stability
Discrimination of Security
Genetic Simulated Annealing Algorithm

Fig. 38. GSS-AODV Structure

5.2.1.2 Node Stability Metrics

Based on the classical AODV protocol algorithm, the relative velocity u and the relative distance d between nodes are extracted, and the node load q is the key factor to measure the node stability. After the above factors are normalized and comprehensively processed, the node stability can be obtained through fuzzy processing to be used in the improved GSS-AODV algorithm.

Set N_i neighbor nodes of node i to form Φ_i collection. j is a node in set Φ_i.

The normalized relative velocity between i and j is defined as formula (5.1). u_{max} is the maximum relative rate between vehicles, such as within the city speed limit $60km/h$, the maximum relative rate of $120km/h$.

$$u_{ij}^{nomal} = \frac{|\overrightarrow{u}_i - \overrightarrow{u}_j|}{u_{max}} \tag{5.1}$$

Suppose the coordinates of node i and neighbor node j are (x_i, y_i), (x_j, y_j), then the relative distance between nodes i and j is defined as formula (5.2).

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{5.2}$$

Then, the normalized relative distance is defined as formula (5.3), where d_{max} is the maximum communication distance, taking $250m$.

$$d_{ij}^{nomal} = \frac{d_{ij}}{d_{max}} \tag{5.3}$$

Suppose the load of neighbor j is q_i, that is, the number of packets stored in the cache queue of this node. Let q_{max} represent the total length of the queue cached by the node, that is, the maximum number of packets allowed to be stored. The normalized load is defined as formula (5.4).

$$q_{ij}^{nomal} = \frac{q_{ij}}{q_{max}} \tag{5.4}$$

5.2.1.3 Node Security Metrics

Several common VANETs internal attacks are: random data packet loss in the process of forwarding; packet tampering and forgery; entice the surrounding nodes to send data packets to malicious nodes to launch black hole attacks. By analyzing these attacks, we can see that when the internal nodes are attacked, the repetition rate of data packets may be too large. When there are attacks such as black hole attacks and selective forwarding, there should be an abnormal number of packets sent[86]. The neighbor table corresponding to a node has a certain correlation, and the neighbor table between normal nodes should be repeated to some extent. Therefore, the packet content repetition rate, the number of packets[87], and the relevance of surrounding nodes can be used as detection factors for malicious nodes in the improved AODV algorithm. Based on the classical AODV protocol algorithm, the detection factors are extracted, normalized, and integrated, and then the node trust value is obtained through the fuzzy processing.

Set N_i neighbor nodes of node i to form Φ_i collection.

As shown in the formula (5.5), $S_{ij}(t)$ represents the normalized packet repetition rate, $T_{ij}(t)$ represents the normalized packet transmission factor, and $U_{ij}(t)$ represents the normalized node similarity.

$$\left\{ \begin{array}{c} S_{ij}(t) = \frac{p_{ij}(t) - sp_{ij}(t)}{P_{ij}(t)} \\ T_{ij}(t) = \frac{|p_{ij}(t) - \Delta p(t)|}{p_{ij}(t)} \\ U_{ij}(t) = \frac{\sum_{c \in N_{ij}(t)} \frac{1}{log_k(c)}}{\sum_{c \in N_{ij}(t)} 1} \end{array} \right\} \tag{5.5}$$

where $p_{ij}(t)$ is the number of packets between i and j at the t moment, $sp_{ij}(t)$ is the number of repeated packets between i and j at the t moment, and the $\Delta p(t)$ is the expected value of the number of packets. $U_{ij}(t)$ is measured by Adamic-Adar[88] indexes. $N_{ij}(t)$ is the intersection of the neighbor set of node i and the neighbor set of node j at the t moment. c is the common neighbor of two nodes, $log_k(c)$ is the logarithm of node degree.

5.2.1.4 Fuzzy Neural Network

In this task, a multi-input and single-output neural network is used. Input normalized relative velocity, the normalized relative distance and the normalized load, with the help of fuzzy neural network, we get the node stability. Similarly, we use fuzzy neural network, the normalized packet repetition rate, the normalized packet transmission factor, and the normalized node similarity to calculate node trust value. Fuzzy Neural Network structure is shown in Fig. 39.

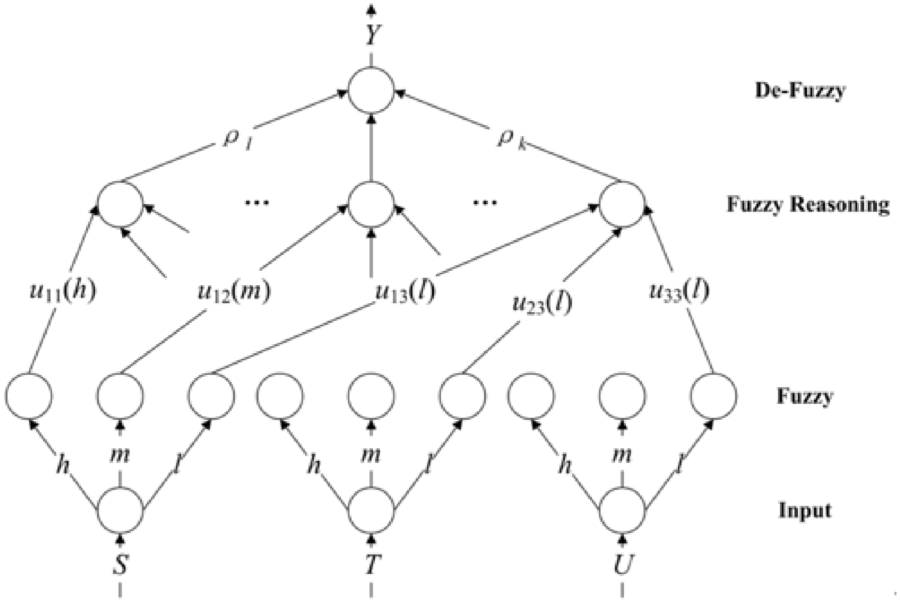


Fig. 39. Fuzzy Neural Network Structure

Take trust value evaluation as an example. The first layer is the input layer, which is responsible for passing the input variables to the second layer. The input value is the exact value. The number of nodes is the number of input variables.

Therefore, this layer has three neuron nodes, also known as three variables, S, T, and U. The second layer is the fuzzy layer, which is mainly to blur the input values. The S, T, and U are converted into three fuzzy subsets $\{high, middleandlow\}$, which can be represented as $\{h, m, l\}$, and there are 9 nodes. u_{ij} means j-th membership fuzzy subset of variables i. In this task, we use the Gaussian function. The third layer is the fuzzy rules reasoning layer. Each node of this layer corresponds to a fuzzy rule, which is connected to the fuzzy subset of every variable in the second layer, and there are 27 nodes, which correspond to 27 rules of the inference. The fourth layer is the defuzzy layer. The fuzzy value of the fuzzy inference is converted into an exact value, and the gravity method is used to blur it, and the output value of the neural network is obtained. ρ_k is the connection weight of the third and fourth layers. Y is the deterministic solution of the problem and the node trust value.

5.2.1.5 Optimizing the Fuzzy Neural Network with Genetic Simulated Annealing Algorithm

In this task, we combined genetic algorithm and simulated annealing algorithm to optimize the parameters used in the fuzzy neural network.

A group of initial population for the global optimal search process is randomly generated. The fitness of each individual is evaluated. A portion of this population is selected to generate a new group of individuals by selection, crossover, mutation and other genetic operations. Then, with the help of simulated annealing algorithm, the individual is fine-tuned to get a higher fitness. The process runs iteratively until some termination condition is satisfied. In summary, there is a thought that the simulated annealing algorithm is dissolved in the running of the genetic algorithm, which not only has the advantages of the genetic algorithm and the simulated annealing algorithm but also overcomes the corresponding deficiencies[89].

5.2.2 GSS-AODV Protocol Description

5.2.2.1 Routing Initiation

When a source node needs to communicate with a destination node, it first performs a route initiation process and broadcasts the RREQ packet to its neighbors. Fig. 40 presents a process flow diagram.

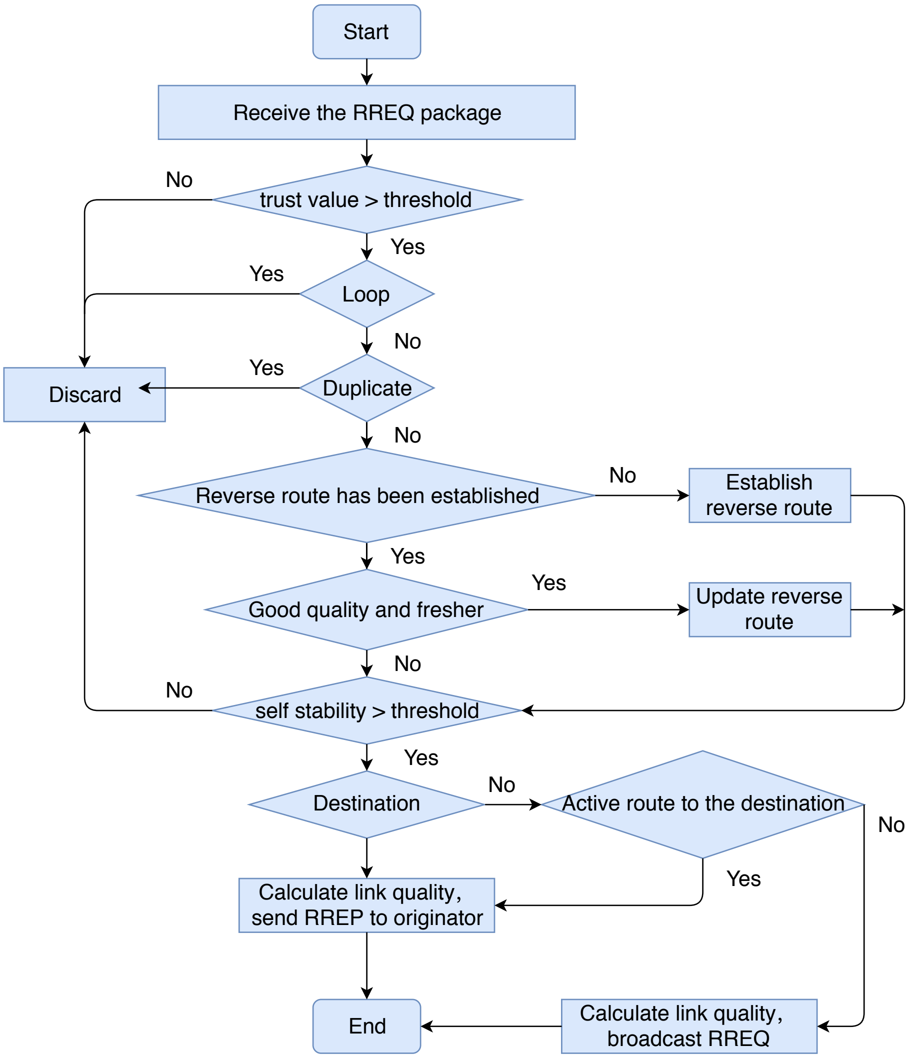


Fig. 40. Routing Initiation Process Flow Diagram

The neighbor nodes that receive the RREQ package perform the following operations in turn:

1) Check for loop, if a loop discards the RREP package.

2) Check for duplicate RREQ packages. If a RREQ package is repeated, discard the RREQ package.

3) Check whether the reverse route to the source node has been established. If the new route has higher quality or similar quality and is fresher than the recorded route, update the last-hop route. Otherwise, a reverse path to the source node should be established. The link quality of the current path is the sum of all nodes' stability encapsulated in the RREQ message divided by the length of route.

4) Set a stable threshold with an initial value of 0.5 in the neighbor node. During a period, the neighbor nodes will use the node stability of the neighbor nodes stored in the neighbor table to calculate the average node stability and update the stability threshold. When the node stability is greater than 0.5 or the stability threshold, the RREQ packet is forwarded. This ensures that stable nodes with stability greater than 0.5 are always able to participate in forwarding. When the node is in the unstable state, it is prevented to participate in forwarding, and the utilization of it is reduced. Therefore, with the help of the stability threshold, it ensures link quality and avoids the entire network forwarding of RREQ messages.

5) Neighbor nodes accumulate the stability and store them in the RREQ packet, then, broadcast RREQ to their neighbor nodes. Different from the AODV protocol, the stability of a node has an active route to the destination still needs to be evaluated. This helps the source node to consider the link quality of the entire path comprehensively. When node stability meets the requirement, the node calculates the link quality from the source node to the destination node and sends an RREP packet containing this link quality to the source node.

6) Every node calculates the average of its neighbors' trust value. Node put the neighbor nodes, whose trust value is smaller than 0.5 or the average neighbor trust values, as in blacklist. They do not participate in the current node routing process. Their messages are ignored.

7)The process ends after the destination node is reached.

5.2.2.2 Routing Choice

RREQ packet is continuously forwarded. Thus, the destination node finally received the RREQ. When the destination node receives the RREQ packet, the node performs a routing process:

1) When the destination node receives the RREQ packet, it first waits for a route discovery period and continuously accepts the RREQ packet before the waiting time.

2)After the waiting time is over, the destination node calculates the link quality according to the cumulative sum of node stability and the number of hops contained in the RREQ packet using formula (5.6), to evaluate the link quality:

$$LQ_m = \frac{St_m}{N} \tag{5.6}$$

Where LQ_m represents the link quality of the m-th link, St_m represents the sum of node stability of the m-th link, and N is the number of link hops. The higher the node stability of the path is and the smaller the number of hops is, the better the link quality is. Compared with AODV, only the path with the smallest hop count is considered, the improved routing protocol can choose a more stable path based on the relatively fewer hops.

3) During the process of receiving, if the path contained in RREQ is fresher or has higher quality than the route in the routing table, the route table should update.

4) When the originator receives an RREP packet, it first checks whether the reverse route of the destination node is established and then determines whether the route needs to be updated by comparing the quality of the link. This process ensures that the reverse path is always stable, then selects the most stable link for data transfer.

5.2.2.3 Routing Maintenance

Nodes periodically send HELLO message to maintain a connection between their neighbor nodes. HELLO message encapsulates node information. Its TTL is 1.

When a node receives a HELLO message from a neighbor node for the first time, it adds the neighbor to its neighbor table and then uses the fuzzy neural network to calculate the node trust value of the corresponding node. It records the repetition rate, the number of packets, the relevance of surrounding nodes, and the average trust values of all the current neighbors in the packet used in the current calculation as training data of the simulated genetic annealing algorithm in the neighbor table.

When a node receives a HELLO message from its neighbor node which is already recorded in the neighbor table, the node first uses the node information encapsulated in the message and the fuzzy neural network to update the node stability and the node trust value, and then prolongs the lifetime of the corresponding neighbor node in the neighbor table.

From time to time, the node checks whether the survival time of all the nodes is less than the current time and considers whether the neighboring node is lost. In the improved routing protocol, when the node is lost, the time difference t_d between the lost time and the received first HELLO message moment is the actual link time with the neighboring node. Calculate the node actually stable St_r by formula (5.7), where MT represents the average node link duration.

$$St_r = \frac{-1}{\left(\frac{t_d}{MT} + 1\right)} + 1 \tag{5.7}$$

This function has a stability of 0.5 at $t_d = MT$ and a positive limit of 1. This St_r is used as the output of training data of genetic simulated annealing algorithm, and the corresponding information of the neighbor table is input as training data to optimize the parameters of the fuzzy neural network. Finally, the lost neighbor node is deleted from the neighbor table. To ensure that the actual stability of the node always meets the current motion environment of the node, the node calculates the average link duration of all the neighboring nodes in the time and updates the MT from time to time.

5.3 Performance Evaluation

In this task, the network simulation software NS2 (Network Simulator Version 2) is used to simulate the improved GSS-AODV protocol and the original AODV protocol.

5.3.1 Stability Experimental Results and Analysis

First, we test the routing performances of GSS-AODV and AODV with different quantity of vehicles when the vehicle speed is $20 \sim 120 m/s$.

Fig. 41 (a) shows the change of delivery rate of AODV and GSS-AODV packets with the increase in the number of vehicles. The GSS-AODV routing protocol finally obtains the link quality by calculating the node stability, which is used to evaluate the link stability. Therefore, the stable link is always selected in the route initiation and selection part to avoid data packet loss. In addition, GSS-AODV can dynamically adjust the parameters used by the fuzzy neural network according to different environments to improve the accuracy of selecting a stable link. Therefore, when the number of nodes is the same, the packet delivery rate of GSS-AODV protocol is always higher than that of AODV protocol, and the relative stability is high.

Fig. 41 (b) shows the end-to-end average delay of AODV and GSS-AODV routing protocols with the increase in the number of nodes. Because the GSS-AODV protocol considers the node load when selecting the link, the node load is taken as a factor to calculate the node stability. Therefore, when the number of nodes is small, GSS-AODV can select a stable and low-load link for data transmission, reduce the queuing time, and reduce the end-to-end delay. When the number of nodes continues to increase, the number of optional paths increases, the load on the nodes generally decreases, and the influence of the node load on the link stability decreases. GSS-AODV will optimize the algorithm parameters through simulated genetic annealing to reduce the weight of the node load in the calculation stability of the fuzzy neural network.

Fig. 41 (c) shows the changes in routing overhead of the two routing protocols, AODV and GSS-AODV, with the increase of the number of nodes. As shown in the figure, with the number of nodes increases, the control overhead of the AODV and GSS-AODV routing protocols also increases. However, the GSS-AODV protocol determines whether to forward the message according to the node stability decision result in the route initiation stage, which limits the flooding of the RREQ message in the network. And GSS-AODV always selects the stable link to transmit data. That link is not easily broken. The number of route repairs is reduced, and the source node does not need to frequently initiate the route, thereby reducing the number of RREQ transmissions.

Then, we test the GSS-AODV protocol routing performance under different vehicle maximum moving speed when the number of vehicles is 100.

Fig. 41 (d) shows the packet delivery rate of AODV protocol and GSS-ADOV protocol with the maximum moving speed of vehicle nodes increases. GSS-AODV routing protocol uses a fuzzy neural network to calculate the node stability. We can get the link quality to evaluate the link stability. In the route initiation and selection part, GSS-AODV always chooses a stable link transmission data, and the link has a lower probability to disconnect. The parameters of the fuzzy neural network are optimized by genetic simulated annealing algorithm under different vehicle speed. The weight of the vehicle speed in the node stability calculation is changed to ensure the stability of the selected link.

Fig. 41 (e) shows the end-to-end average delay of AODV protocol and GSS-AODV protocol with the increase in the maximum moving speed of vehicles. At low speed, because GSS-AODV waits for a route discovery cycle, it will cause some network delay. However, early restoration is added during route restoration to avoid the delay caused by packet loss. At high speed, the GSS-AODV protocol always selects the stable link to transmit data, which can reduce the delay caused by the route repair.

Fig. 41 (f) shows the routing overhead of AODV protocol and GSS-AODV protocol with the increase in the maximum moving speed of vehicles. GSS-AODV controls the forwarding of RREQ by the unstable node in the route initiation part and adjusts the weight of movement speed in the fuzzy neural network according to the motion condition and the parameters controlled by genetic simulated annealing algorithm. The GSS-AODV can select stable links to transmit data at different speeds to reduce the routing overhead required by rerouting the links when the links are disconnected.

5.3.2 Security Experimental Results and Analysis

First, we test the routing performance of the GSS-AODV protocol under the different number of black hole nodes, when the number of vehicles is 100.

Fig. 42 (e) shows the packet loss ratio of AODV protocol and GSS-ADOV protocol with the number of black holes increase. As the figure shows, with the increase in the number of black hole nodes, more REEQ packets are phagocytic, and the loss rate of the two protocols increases. The GSS-AODV routing protocol uses fuzzy neural network to calculate the node trust value and select the nodes with high trust value to participate in the routing initiation. It reduces the probability of attack and the number of nodes in different situations by genetic simulated annealing algorithm, so as to optimize the parameters of fuzzy neural network and change the node correlation in the node trust value weight calculation the increased probability of select safe node.

Fig. 42 (a) shows the end-to-end average delay of AODV and GSS-AODV with the increase in the number of black hole nodes. In the environment with fewer black hole nodes, the GSS-AODV protocol preferentially selects the nodes with higher trust values to participate in the routing process, which causes certain network delay. However, routing protocol optimizes the parameters according to the specific conditions and avoids prolonged delay, so that the average delay does not show a large gap. With the increase of black hole nodes quantity, the GSS-AODV protocol always selects the nodes with higher trust values to participate in the routing process and reduces the probability of routing requests being swallowed by the attacking nodes.

Fig. 42 (c) shows the routing overhead of AODV and GSS-AODV with the increase in the number of black hole nodes. When the number of black hole nodes increases, both the probability of losing RREQ and the number of control messages between nodes increase. Therefore, the routing load increases. However, since the GSS-AODV controls the forwarding of the RREQ through the node trust value in the routing initiation part and uses the genetic simulated annealing algorithm to control the parameters, the weight of the node similarity in the fuzzy neural network is adjusted according to the condition of the black hole node. In different environments, GSS-AODV can select the secure node to participate in the routing process to reduce the routing overhead required for initiating the route initiation due to the black hole node attacks.

We also test the routing performance of the GSS-AODV protocol under different numbers of vehicle nodes, when the ratio of the black hole nodes is ten percent.

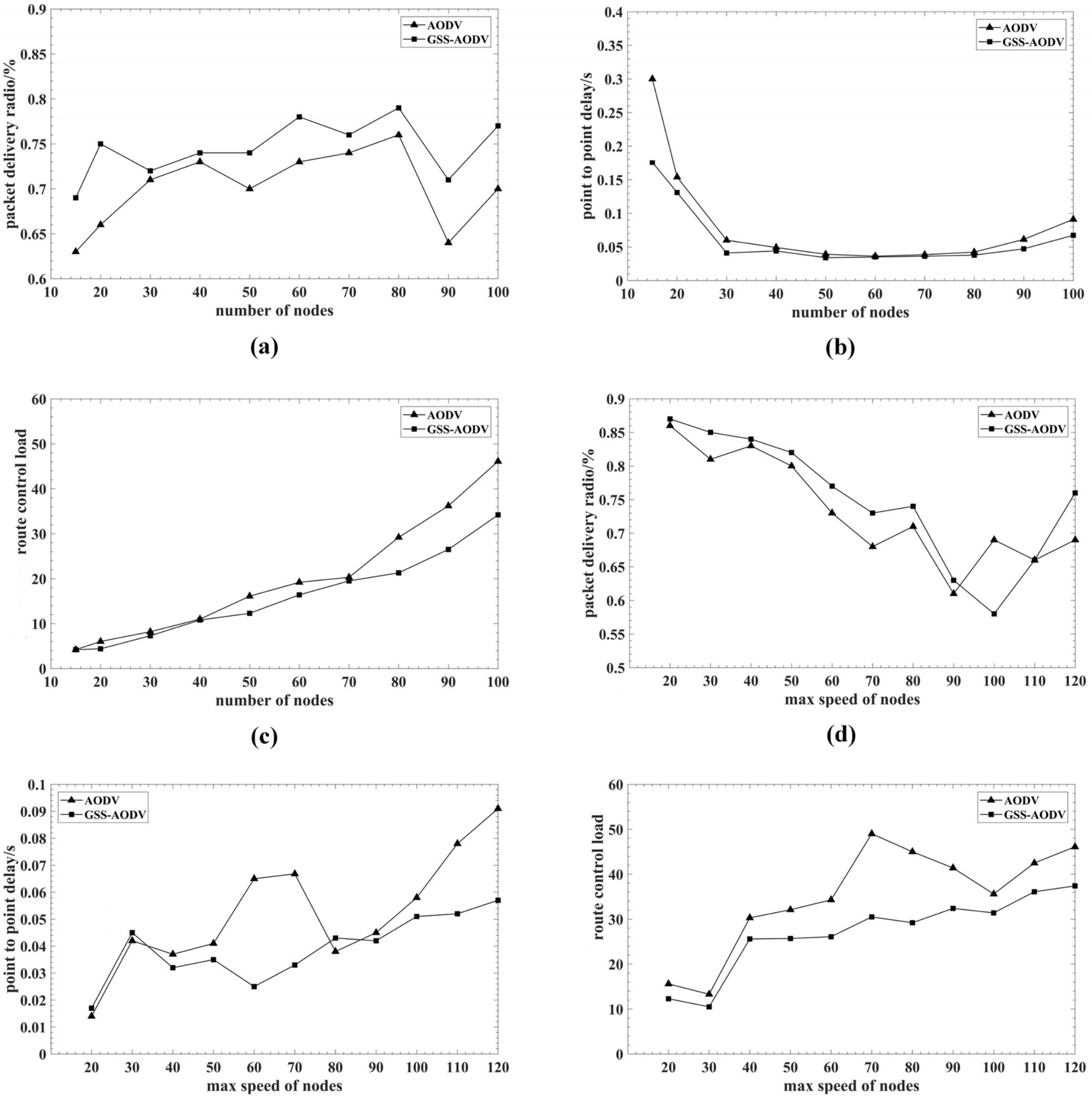
Fig. 42 (f) shows the packet loss rate of AODV and GSS-AODV with the increase in the number of nodes. GSS-AODV routing protocol evaluates the node security by calculating the node trust value. Therefore, routing protocol always selects relatively secure nodes to participate in the routing process to reduce the impact of black hole nodes in node communication and increase the packet delivery rate. Also, GSS-AODV can dynamically adjust the parameters used by the fuzzy neural network according to different environments to improve the accuracy of selecting a safe node.

Fig. 42 (b) shows the end-to-end average delay of two routing protocols, AODV and GSS-AODV, with the increase in the number of nodes. When selecting a node to participate in the routing process, the GSS-AODV protocol takes the amount of data packets sent and the repetition rate as influence factors into the calculation of the trust value of the node. When the number of nodes is small, the GSS-AODV protocol mainly considers the packet repetition rate and improves the weight of the packet repetition rate in the node trust value, which helps the routing protocol to prevent the attack node from repeatedly sending attack messages and affecting the ad hoc network.

Fig. 42 (d) shows how the routing costs of AODV and GSS-AODV routing protocols changes with the increase of the number of nodes. In the route initiation phase, GSS-AODV protocol determines whether to participate in the routing process according to the judgment result of the node trust value. Thereby it limits the flooding of the RREQ message in the network. And since the GSS-AODV always selects the secure node to transmit data, the data packet is not easy to be lost, and the routing security is enhanced. Therefore, the source node does not need to initiate routing frequently, and the number of RREQ transmission is reduced.

5.4 Conclusion

Stability and security are both hot issues in VANET research. This paper presents a stable AODV routing algorithm, named GSS-AODV, based on the fuzzy neural network. In GSS-AODV, the node stability and route length are considered in equilibrium, and the parameters are adjusted by genetic simulated annealing algorithm under different practical conditions to ensure that the calculated node stability is in accordance with the actual situation. The proposed algorithm takes into consideration of multiple attack models and adjusts the parameters through genetic simulated annealing algorithm in different practical environments to improve the accuracy of node trust value. Experimental results show that GSS-AODV can choose the route including more secure nodes, reduce the packet loss rate, reduce the average end-to-end delay, and normalize routing overhead.



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 41. Stability Experimental Results and Analysis

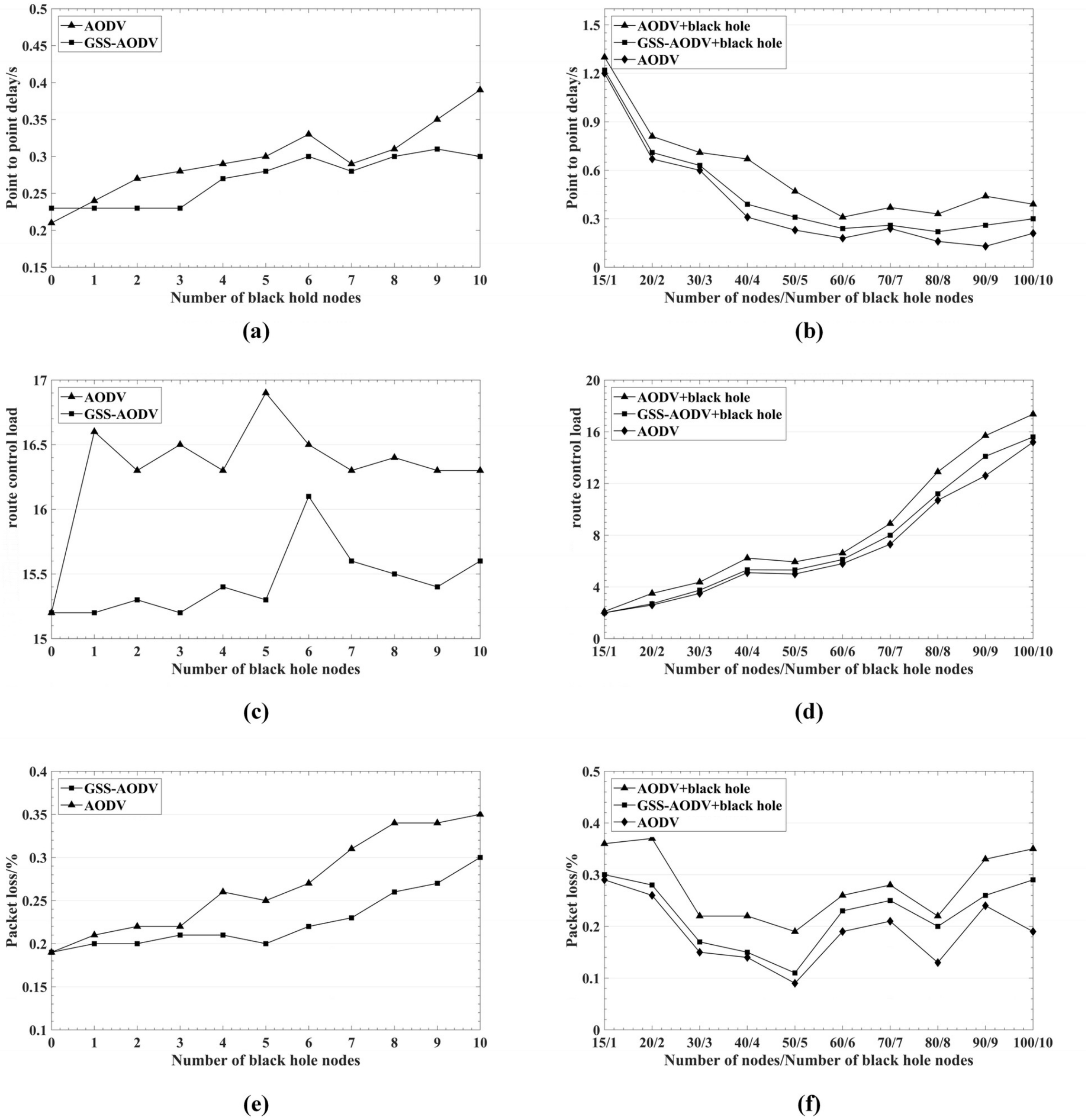


Fig. 42. Security Experimental Results and Analysis

CHAPTER 6

TASK II.2: MULTI-ANTENNA BASED INFORMATION PRIVACY PROTECTION FOR V2V COMMUNICATIONS

6.1 Problem Statement

As vehicles meet each other occasionally in vehicular networks, the passenger in a vehicle may just want to have a one-time conversation with its neighbors. Consequently, the user will have the requirement of protecting its privacy via (a) not revealing its IDs (such as user name) to its neighbors even during the process of connection setup and (b) making the application servers (which could be the road-side base stations) unaware of who participate in the conversations. This anonymity requirement brings challenges to the research for the connection setup and the connection maintenance during a conversation.

Some related work and previous mechanisms are presented in **Chapter 2**. Some proposed communication approaches are based on geo-location.Some previous network security research focuses on protocol and IP address. Our goal is to propose a novel approach which can meet the anonymous requirements in vehicular networks.

To meet the anonymous requirements, considering the feasibility of installing multiple antennas on vehicles, we propose to utilize physical layer information (RSS-Ratio), which have the properties such as (i) unique for a user, (ii) frequently updated, (iii) including location information, and (iv) hard to be obtained by attackers, to generate temporary IDs for the communication parties.

Received Signal Strength (RSS) is an indicator of the signal strength in receiver for wireless communications. Ideally, the received signal strength can be quantified as follows[34]:

$$P_r[dBM] = P_0 - 20\log(\frac{d}{d_0}) \tag{6.1}$$

where p_0 is the signal power at distance d_0 away from the sender, d is the distance between receive and send antennas.

Since RSS is susceptible to environmental changes, considering the fading caused by environment, a log-normal shadowing model is widely used to analyze RSS:

$$P_r[dBM] = P_0 - 20\alpha\log(\frac{d}{d_0}) + \chi_\sigma \tag{6.2}$$

where α is the path loss exponent, χ_σ is background noise, which is a Gaussian distributed random variable with zero mean and standard deviation σ.

All environmental unknowns make it difficulty to obtain an accurate RSS value. To eliminate environmental unknowns, [90] proposes a new definition *RSS-Ratio* denoted by τ, is the output value of the three-antenna based RSS processing. Formally,:

$$\tau = \frac{P_r^1 - P_r^2}{P_r^1 - P_r^3} = \frac{log(\frac{d_1}{d_2})}{log(\frac{d_1}{d_3})} \tag{6.3}$$

where P_r^1, P_r^2, and P_r^3 are respectively the RSS measured at the receiver's antenna $\sharp 1$, $\sharp 2$, and $\sharp 3$; and d_1, d_2 , and d_3 , are respectively the distance from the signal source to the receiver's antenna $\sharp 1$, $\sharp 2$, and $\sharp 3$.

6.2 Proposed Approach

6.2.1 RSS-Ratio Properties Analysis

We propose an approach to utilize RSS-Ratio for implementing anonymous communications in vehicular networks.

Since RSS-Ratio only depends on distance information, it is much stabler than RSS. Fig. 43 demonstrates the values of RSS and RSS-Ratio under several trans-

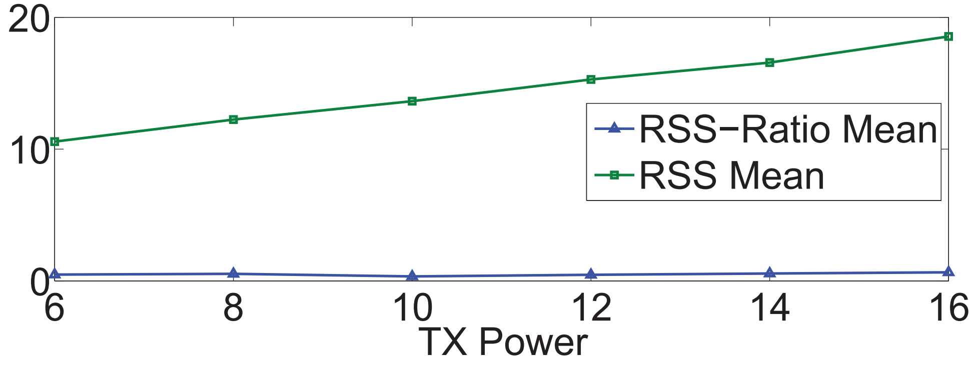


Fig. 43. Impact of Transmission Power on RSS and RSS-Ratio

mission powers in outdoor environments. One can see that RSS-Ratio is much more stable than RSS over different transmission powers. Similar results have also been observed when time and other environmental unknowns vary.

Moreover, from Eqn. (6.3), we can see that RSS-Ratio is theoretically only related to the distances from the signal source to the receiver's three antennas. It is, therefore, predicable given the distances. The geographical distribution of RSS-Ratio value has interesting features. Given an RSS-Ratio value and three receiving antennas deployed at (0,0), (1,0), and (0, 1), almost all the possible locations of the senders, which can produce the given RSS-Ratio value at the receiver, appear on a straight line [90] as shown in Fig. 44, where the number beside a line is the given RSS-Ratio value and the line is the possible locations of the senders.

From Fig. 44, we have two interesting observations: (i) the value of RSS-Ratio is relatively stable as long as the relative position between the two vehicles (the sender and the receiver) does not change too much; (ii) even more, as the relative position changes slowly (comparing to the time of signal transmission and RSS-Ratio computation), the RSS-Ratio of the sender at the next time instance is predictable without the need of knowing the physical locations of the two vehicles. It is, therefore,

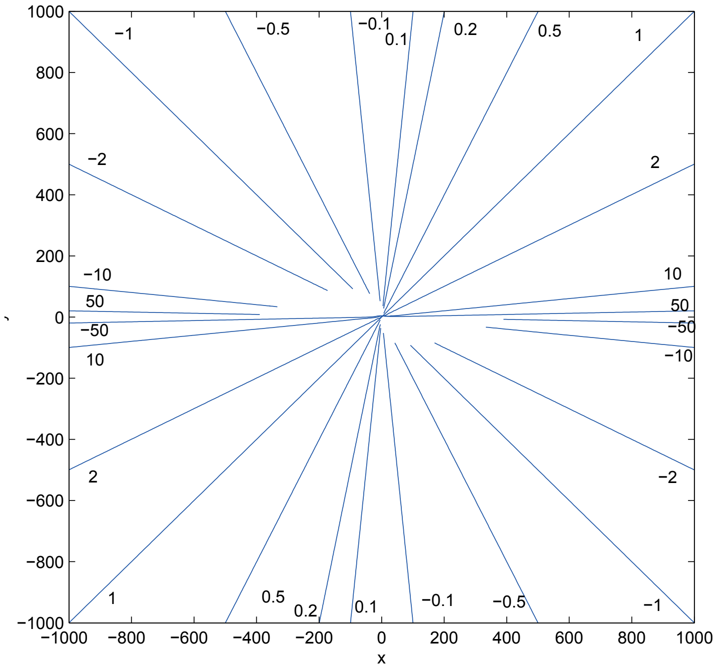


Fig. 44. RSS-Ratio Distribution

possible to utilize RSS-Ratio for connection setup and maintenance. In addition, other vehicles that are not at the receiver's position and may not have the same deployment of the three receiving antennas as the one at the receiver, are not able to estimate the distances d_1, d_2, and d_3 from the signal source to the receiver's antennas. As a result, they will not be able to have the same RSS-Ratio for the same signal source. In other words, the RSS-Ratio for one pair of source and destination is unique.

6.2.2 Anonymous Connection Setup and Maintenance

Based on theses analytical results, we propose to use RSS-Ratio as the address for the following reasons:

(i) RSS-Ratio is a value related to the antenna layout and the relative locations of the sender and the receiver. It is very rare for the receiving vehicle to have the same measured RSS-Ratio from two vehicles;

(ii) RSS-Ratio changes relatively slowly when the vehicles are moving together;

(iii) It is almost impossible to predict a RSS-Ratio by attackers as they do not know the exact receiving antenna layout and the exact relative locations. Note that even the sender cannot predict its RSS-Ratio measured by the receiver.

As the measured RSS-Ratio is time sensitive, the receiver can have a series of measured RSS-Ratio values from the sender. Note that these RSS-Ratio values are also and only predicable by the receiver. We will use these RSS-Ratio values to identify and maintain a TPC connection. The connection setup algorithm is depicted in Fig. 44. Particularly, we propose to follow the steps below for connection setup and maintenance.

Steps of Connection Setup and Maintenance

1) The sender randomly sends several preambles to the receiver so that the receiver can measure the RSS-Ratio for each preamble.

2) The sender sends a connection setup request to the receiver with its location information, where the preambles were sent. Note that the sender will add several faked locations as well.

3) The receiver measures the RSS-Ratio for each preamble and the connection request packet. As the receiver can calculate the RSS-Ratio based on the sender's location and its own receiving antenna layout, it can use fuzzy vault scheme to find out which are the sender's locations.

4) The receiver uses these measured RSS-Ratio values as the address of the sender and repeats step-(1-3) as a new sender so that the sender in step-(1) can set up the connection from its side.

5) When sending each packet, the sender adds its location and several faked locations. The receiver uses the method in step-(3) to maintain the connection.

Note that the connection setup and maintenance procedure step-(1-5) is also

```
Algorithm 1: Anonymous Connection Setup
Input: vehicle $s$, vehicle $r$, location information $ls$ of where $s$ sent
       preambles, location information $lr$ of where $r$ sent preambles
if $s$ wants to connect to $r$ then
    Randomly send several preambles to r;
    Send a connection setup request to $r$ with $ls$ and several faked
      locations;
    if $r$ wants to be connected to then
        Measure the RSS-Ratio for each preamble and the connection
          request packet;
        Get $ls$;
        Randomly send several preambles to $s$;
        Send a connection setup request to s with $lr$ and several faked
          locations;
    end
    $s$ measures the RSS-Ratio for each preamble and the connection
      request packet;
    Get $lr$;
    Setup connection;
end
```

Fig. 45. Anonymous Connection Setup Algorithm

the key generation and exchange procedure by using the sender's locations as the key seed. Moreover, by following the proposed steps, an attacker cannot pretend to be the sender as its measured RSS-Ratio values are not consistent with the ones measured from the sender, and cannot decrypt a message as it does not know which are the sender's real locations.

6.3 Conclusion

As vehicles meet each other occasionally in vehicular network, users will naturally have the requirement of protecting their privacy during vehicular network communications. The privacy requirements bring unique challenges for the communication setup and maintenance as any information (such as IP address and MAC address) that may relate to the users' IDs should not be used. To resolve this problem, we propose to utilize the dynamic physical level information, RSS-Ratio, as the address.

We briefly present why RSS-Ratio can be used as an address while not revealing users' IDs, and our proposed method of utilizing RSS-Ratio for anonymous connection setup and maintenance.

REFERENCES

[1] Kashif Naseer Qureshi and Abdul Hanan Abdullah. “A survey on intelligent transportation systems”. In: *Middle-East Journal of Scientific Research* 15.5 (2013), pp. 629–642.

[2] Saif Al-Sultan et al. “A comprehensive survey on vehicular ad hoc network”. In: *Journal of network and computer applications* 37 (2014), pp. 380–392.

[3] MS Kakkasageri and SS Manvi. “Information management in vehicular ad hoc networks: A review”. In: *Journal of Network and Computer Applications* 39 (2014), pp. 334–350.

[4] Richard Gilles Engoulou et al. “VANET security surveys”. In: *Computer Communications* 44 (2014), pp. 1–13.

[5] Santosh Chokhani. “Toward a national public key infrastructure”. In: *IEEE Communications Magazine* 32.9 (1994), pp. 70–74.

[6] Yi-Shiung Yeh, Wei-Shen Lai, and Chung-Jaye Cheng. “Applying lightweight directory access protocol service on session certification authority”. In: *Computer Networks* 38.5 (2002), pp. 675–692.

[7] T Perlines Hormann, Konrad Wrona, and Silke Holtmanns. “Evaluation of certificate validation mechanisms”. In: *Computer Communications* 29.3 (2006), pp. 291–305.

[8] Patrick T Eugster et al. “Epidemic information dissemination in distributed systems”. In: *Computer* 37.5 (2004), pp. 60–67.

[9] Maziar Nekovee and Benedikt Bjarni Bogason. “Reliable and effcient information dissemination in intermittently connected vehicular adhoc networks”. In: *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*. IEEE. 2007, pp. 2486–2490.

[10] Jason J Haas, Yih-Chun Hu, and Kenneth P Laberteaux. “Efficient certificate revocation list organization and distribution”. In: *IEEE Journal on Selected Areas in Communications* 29.3 (2011), pp. 595–604.

[11] Patricia Ruiz et al. “Information dissemination in VANETs based upon a tree topology”. In: *Ad Hoc Networks* 10.1 (2012), pp. 111–127.

[12] PB Farradyne. *Vehicle Infrastructure Integration-VII Architecture and Functional Requirement*. `http://ral.ucar.edu/projects/vii.old/vii/docs/VIIArchandFuncRequirements.pdf`. April 8, 2005.

[13] Suvadip Batabyal and Parama Bhaumik. “Delay-overhead trade-offs in mobile opportunistic network using TTL based restricted flooding”. In: *Applications and Innovations in Mobile Computing (AIMoC), 2014*. IEEE. 2014, pp. 9–14.

[14] DN Rewadkar and Mithari Priti Madhukar. “An adaptive routing algorithm using dynamic TTL for data aggregation in Wireless Sensor Network”. In: *Current Trends in Engineering and Technology (ICCTET), 2014 2nd International Conference on*. IEEE. 2014, pp. 192–197.

[15] Jiaxin Ding, Jie Gao, and Hui Xiong. “Understanding and modelling information dissemination patterns in vehicle-to-vehicle networks”. In: *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM. 2015, p. 41.

[16] Felipe D Cunha et al. “Are vehicular networks small world?” In: *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on*. IEEE. 2014, pp. 195–196.

[17] Hong Zhang and Jie Li. “Modeling and dynamical topology properties of VANET based on complex networks theory”. In: *AIP Advances* 5.1 (2015), p. 017150.

[18] Xinjing Wang et al. “An urban area-oriented traffic information query strategy in VANETs”. In: *International Conference on Wireless Algorithms, Systems, and Applications*. Springer. 2013, pp. 313–324.

[19] Guillaume Remy et al. “LTE4V2X: LTE for a centralized VANET organization”. In: *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*. IEEE. 2011, pp. 1–6.

[20] Agon Memedi et al. “Cluster-based transmit power control in heterogeneous vehicular networks”. In: *Vehicular Networking Conference (VNC), 2015 IEEE*. IEEE. 2015, pp. 60–63.

[21] Yujin Li, Ming Zhao, and Wenye Wang. “Intermittently connected vehicle-to-vehicle networks: detection and analysis”. In: *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*. IEEE. 2011, pp. 1–6.

[22] Ping Dong et al. “Energy-efficient cluster management in heterogeneous vehicular networks”. In: *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*. IEEE. 2016, pp. 644–649.

[23] S Louis Hakimi. “Optimum distribution of switching centers in a communication network and some related graph theoretic problems”. In: *Operations Research* 13.3 (1965), pp. 462–475.

[24] Quyuan Luo et al. “CFT: A cluster-based file transfer scheme for highway VANETs”. In: *Communications (ICC), 2017 IEEE International Conference on*. IEEE. 2017, pp. 1–6.

[25] C Perkins, E Belding-Royer, and S Das. *Request for Comments: Ad hoc ondemand distance vector (AODV) routing*. 2003.

[26] Apurva Jain, Urmila Prajapati, and Piyush Chouhan. “Trust based mechanism with AODV protocol for prevention of black-hole attack in MANET scenario”. In: *Colossal Data Analysis and Networking (CDAN), Symposium on*. IEEE. 2016, pp. 1–4.

[27] H Ushikubo, S Takeda, and H Shigeno. “An effective secure routing protocol considering trust in mobile ad hoc networks”. In: *IPSJ Journal* 55.2 (2014), pp. 649–658.

[28] Sayaka Umeda, Sonoko Takeda, and Hiroshi Shigeno. “Trust evaluation method adapted to node behavior for secure routing in mobile ad hoc networks”. In: *Mobile Computing and Ubiquitous Networking (ICMU), 2015 Eighth International Conference on*. IEEE. 2015, pp. 143–148.

[29] Guoxing Zhan, Weisong Shi, and Julia Deng. “Design and implementation of TARF: A trust-aware routing framework for WSNs”. In: *IEEE Transactions on dependable and secure computing* 9.2 (2012), pp. 184–197.

[30] Huaizhi Li and Mukesh Singhal. “A secure routing protocol for wireless ad hoc networks”. In: *System Sciences, 2006. HICSS’06. Proceedings of the 39th Annual Hawaii International Conference on*. Vol. 9. IEEE. 2006, 225a–225a.

[31] Jacek Rak. “Providing differentiated levels of service availability in VANET communications”. In: *IEEE Communications Letters* 17.7 (2013), pp. 1380–1383.

[32] Jacek Rak. “LLA: A new anypath routing scheme providing long path lifetime in VANETs”. In: *IEEE communications letters* 18.2 (2014), pp. 281–284.

[33] Akram Kout, Said Labed, Salim Chikhi, et al. “AODVCS, a new bio-inspired routing protocol based on cuckoo search algorithm for mobile ad hoc networks”. In: *Wireless Networks* (2017), pp. 1–11.

[34] Yang Wu et al. “R2NA: Received Signal Strength (RSS) ratio-based node authentication for body area network”. In: *Sensors* 13.12 (2013), pp. 16512–16532.

[35] Liang Cai et al. “Good neighbor: Secure pairing of nearby wireless devices by multiple antennas”. In: *Proceedings of Network and Distributed Systems Security Symposium*. 2011.

[36] Timothy J Pierson et al. “Wanda: securely introducing mobile devices”. In: *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*. IEEE. 2016, pp. 1–9.

[37] Xiaodong Lin et al. “Security in vehicular ad hoc networks”. In: *IEEE communications magazine* 46.4 (2008).

[38] Vimal Bibhu et al. “Performance analysis of black hole attack in VANET”. In: *International Journal Of Computer Network and Information Security* 4.11 (2012), p. 47.

[39] Dinesh Goyal. “Design and Analysis of Secure VANET Framework preventing Black Hole and Gray Hole Attack”. In: (2016).

[40] Nitish Shukla et al. “Security in vehicular ad hoc network by using multiple operating channels”. In: *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on.* IEEE. 2016, pp. 3064–3068.

[41] Philippe Golle, Dan Greene, and Jessica Staddon. “Detecting and correcting malicious data in VANETs”. In: *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks.* ACM. 2004, pp. 29–37.

[42] Matthias Gerlach and Felix Guttler. “Privacy in VANETs using changing pseudonyms-ideal and real”. In: *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th.* IEEE. 2007, pp. 2521–2525.

[43] Krishna Sampigethaya et al. “AMOEBA: Robust location privacy scheme for VANET”. In: *IEEE Journal on Selected Areas in Communications* 25.8 (2007).

[44] Mike Burmester, Emmanouil Magkos, and Vassilis Chrissikopoulos. “Strengthening privacy protection in vanets”. In: *Networking and Communications, 2008. WIMOB’08. IEEE International Conference on Wireless and Mobile Computing,* IEEE. 2008, pp. 508–513.

[45] Tat Wing Chim et al. “VSPN: VANET-based secure and privacy-preserving navigation”. In: *IEEE Transactions on Computers* 63.2 (2014), pp. 510–524.

[46] Xiaodong Lin et al. “STAP: A social-tier-assisted packet forwarding protocol for achieving receiver-location privacy preservation in VANETs”. In: *INFOCOM, 2011 Proceedings IEEE.* IEEE. 2011, pp. 2147–2155.

[47] Wenjing Wang, Huapu Lu, and Jie Zhang. “A vehicle management system of community based on radio frequency identification technology”. In: *ICLEM 2010: Logistics For Sustained Economic Development: Infrastructure, Information, Integration.* 2010, pp. 1619–1625.

[48] Di Liu. “An intelligent Communication terminal for the Campus-Vehicle Early Warning System Based on RFID”. In: *Procedia Engineering* 15 (2011), pp. 2545–2549.

[49] Jianqiang Wang, Daiheng Ni, and Keqiang Li. “RFID-based vehicle positioning and its applications in connected vehicles”. In: *Sensors* 14.3 (2014), pp. 4225–4238.

[50] Anala Aniruddha Pandit, Ankit Kumar Mundra, and Jyot Talreja. “RFID tracking system for vehicles (RTSV)”. In: *Computational Intelligence, Communication Systems and Networks, 2009. CICSYN’09. First International Conference on*. IEEE. 2009, pp. 160–165.

[51] Wang Hongjian and Tang Yuelin. “RFID technology applied to monitor vehicle in highway”. In: *Digital Manufacturing and Automation (ICDMA), 2012 Third International Conference on*. IEEE. 2012, pp. 736–739.

[52] Soichi Kubota, Yoshiharu Okamoto, and Hideo Oda. “Safety driving support system using RFID for prevention of pedestrian-involved accidents”. In: *ITS Telecommunications Proceedings, 2006 6th International Conference on*. IEEE. 2006, pp. 226–229.

[53] Aritra Paul et al. “An RFID based in-vehicle alert system for road oddities”. In: *Recent Advances in Intelligent Computational Systems (RAICS), 2011 IEEE*. IEEE. 2011, pp. 019–024.

[54] Kashif Ali and Hossam Hassanein. “Using passive RFID tags for vehicle-assisted data dissemination in intelligent transportation systems”. In: *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*. IEEE. 2009, pp. 688–694.

[55] Kirti Chawla et al. “Real-time RFID localization using RSS”. In: *Localization and GNSS (ICL-GNSS), 2013 International Conference on*. IEEE. 2013, pp. 1–6.

[56] Zeyuan Li, Pei-Jung Chung, and Bernard Mulgrew. “Distributed target localization using quantized received signal strength”. In: *Signal Processing* 134 (2017), pp. 214–223.

[57] Lina Altoaimy, Imad Mahgoub, and Monika Rathod. “Weighted localization in vehicular ad hoc networks using vehicle-to-vehicle communication”. In: *Global Information Infrastructure and Networking Symposium (GIIS), 2014*. IEEE. 2014, pp. 1–5.

[58] Lina Altoaimy and Imad Mahgoub. “OWL: optimized weighted localization for vehicular ad hoc networks”. In: *Connected Vehicles and Expo (ICCVE), 2014 International Conference on*. IEEE. 2014, pp. 699–704.

[59] P Farradyne. “Vehicle infrastructure integration (VII)-architecture and functional requirements”. In: *Draft Version* 1 (2005).

[60] Baohua Huang et al. “Optimizing propagation network of certificate revocation in VANET with meet-table”. In: *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*. Springer. 2016, pp. 147–154.

[61] Giorgia V Rossi et al. “Stable clustering for ad-hoc vehicle networking”. In: *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*. IEEE. 2017, pp. 1–6.

[62] Zaydoun Y Rawashdeh and Syed Masud Mahmud. “A novel algorithm to form stable clusters in vehicular ad hoc networks on highways”. In: *EURASIP Journal on Wireless Communications and Networking* 2012.1 (2012), p. 15.

[63] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. “Understanding individual human mobility patterns”. In: *Nature* 453.7196 (2008), pp. 779–782.

[64] Valery Naumov, Rainer Baumann, and Thomas Gross. “An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces”. In: *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*. ACM. 2006, pp. 108–119.

[65] SFMTA. *SAN FRANCISCO TRANSPORTATION FACT SHEET*. December 30, 2013.

[66] Will Reisman. *Commute speeds have slowed down for San Francisco drivers*. `http://archives.sfexaminer.com/sanfrancisco/commute-speeds-have-slowed-down-for-san-francisco-drivers/Content?oid=2187521`. Accessed December 6, 2011.

[67] Xingbang Tian, Baohua Huang, and Min Wu. “A transparent middleware for encrypting data in MongoDB”. In: *Electronics, Computer and Applications, 2014 IEEE Workshop on*. IEEE. 2014, pp. 906–909.

[68] Lei Yu et al. “Cooperative end-to-end traffic redundancy elimination for reducing cloud bandwidth cost”. In: *Network Protocols (ICNP), 2012 20th IEEE International Conference on*. IEEE. 2012, pp. 1–10.

[69] Emmanuel Ndashimye et al. “Vehicle-to-infrastructure communication over multi-tier heterogeneous networks: A survey”. In: *Computer Networks* 112 (2017), pp. 144–166.

[70] Tarek Bouali, Sidi-Mohammed Senouci, and Hichem Sedjelmaci. “A distributed detection and prevention scheme from malicious nodes in vehicular networks”. In: *International Journal of Communication Systems* 29.10 (2016), pp. 1683–1704.

[71] Flavio Bonomi et al. “Fog computing and its role in the internet of things”. In: *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM. 2012, pp. 13–16.

[72] Jun Luo and Jean-Pierre Hubaux. *A survey of inter-vehicle communication*. Tech. rep. 2004.

[73] Yunxin Jeff Li. “An overview of the DSRC/WAVE technology”. In: *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. Springer. 2010, pp. 544–558.

[74] Kan Zheng et al. “Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions”. In: *IEEE Communications Surveys & Tutorials* 17.4 (2015), pp. 2377–2396.

[75] Tony Lindeberg. “Feature detection with automatic scale selection”. In: *International journal of computer vision* 30.2 (1998), pp. 79–116.

[76] Christoph Sommer et al. “A computationally inexpensive empirical model of IEEE 802.11 p radio shadowing in urban environments”. In: *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*. IEEE. 2011, pp. 84–90.

[77] Michael Behrisch et al. “SUMO–simulation of urban mobility: an overview”. In: *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind. 2011.

[78] OpenStreetMap Foundation (OSMF). *OpenStreetMap*. URL: `http://www.openstreetmap.org/#map=14/38.9004/-77.0282`.

[79] Edsger W Dijkstra. “A note on two problems in connexion with graphs”. In: *Numerische mathematik* 1.1 (1959), pp. 269–271.

[80] James MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[81] Elyes Ben Hamida and Muhammad Awais Javed. “Channel-aware ECDSA signature verification of basic safety messages with k-means clustering in VANETs”. In: *Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference on*. IEEE. 2016, pp. 603–610.

[82] Qingwei Zhang et al. “An efficient certificate revocation validation scheme with k-means clustering for vehicular ad hoc networks”. In: *Computers and Communications (ISCC), 2012 IEEE Symposium on*. IEEE. 2012, pp. 000862–000867.

[83] Rong Chai, Xianlei Ge, and Qianbin Chen. “Adaptive K-Harmonic Means clustering algorithm for VANETs”. In: *Communications and Information Technologies (ISCIT), 2014 14th International Symposium on*. IEEE. 2014, pp. 233–237.

[84] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. *Ad hoc on-demand distance vector (AODV) routing*. Tech. rep. 2003.

[85] Jerome Haerri, Fethi Filali, and Christian Bonnet. “Performance comparison of AODV and OLSR in VANETs urban environments under realistic mobility patterns”. In:

[86] Bikramjeet Singh, Dasari Srikanth, and CR Suthikshn Kumar. “Mitigating effects of Black hole Attack in Mobile Ad-Hoc NETworks: Military perspective”. In: *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*. IEEE. 2016, pp. 810–814.

[87] Yansong Zhu and Guiqin Dou. “A dimensional trust based security data aggregation method in wireless sensor networks”. In: *Journal of Wuhan University(Natural Science Edition)* 59.2 (2013), pp. 193–197.

[88] Lada A Adamic and Eytan Adar. “Friends and neighbors on the web”. In: *Social networks* 25.3 (2003), pp. 211–230.

[89] Xiao Wu et al. “Synthesis of Large-scale Multistream Heat Exchanger Networks Based on Stream Pseudo Temperature1”. In: *Chinese Journal of Chemical Engineering* 14.5 (2006), pp. 574–583.

[90] Wei Cheng et al. “RSS-Ratio for enhancing performance of RSS-based applications”. In: *INFOCOM, 2013 Proceedings IEEE*. IEEE. 2013, pp. 3075–3083.

VITA

Xiaolu Cheng graduated with a Bachelor of Science in Electronic Information Science and Technology from Shandong University of Science and Technology, China. After receiving her B.S., she started her Ph.D. study in Computer Science at Virginia Commonwealth University under the direction of Dr. Wei Cheng. Her research interests include vehicular networks and network security. She has won Best Paper Award at the international conference on Identification, Information and Knowledge in the Internet of Things (IIKI) 2017.