Design and Evaluation of Cooperative Location Verification Protocol for Vehicular Ad-Hoc Networks

by

Pengfei Zhang

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.A.Sc. degree in
Electrical and Computer Engineering

School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

Abstract

Vehicular ad hoc networks (VANETs) have attracted much attention over the last few years. VANETs own several significant characteristics, such as the high-rate changing topology led by velocity of vehicles, time-and-location critical safety applications, and Global Positioning System (GPS) devices.

In VANETs, as vehicle movement is usually restricted in just bidirectional movements along the roads and streets, geographical location information becomes very useful. In addition, many studies show that position-based routing protocol is a more promising routing strategy for VANETs; therefore security and verification of location information are necessary to be researched.

In this thesis, a location verification approach, namely the Cooperative Location Verification (CLV) protocol, is proposed, aiming to prevent position-spoofing attacks on VANETs. The CLV basically uses two vehicles, a Verifier and a Cooperator, to verify the claimed position of a vehicle (Prover), according to two challenge-response procedures. Additionally, the security analysis of the CLV is presented.

In order to enhance the CLV by reduce the network overhead, a reputation management system is designed. It utilizes the verification results of the CLV application and maintain every vehicle's reliability in the network. In addition, the solution to sparse networks is briefly discussed.

In the simulation, the results show that the proposed CLV performs better than another location verification algorithm, namely the Secure Location Verification (SLV). And the effectiveness of the reputation management system is also demonstrated.

Acknowledgements

I am deeply indebted to my thesis supervisor: Dr. Azzedine Boukerche, who approved my topic, edited my thesis and gave me invaluable advice both in person and by e-mail. I would also like to thank him for his wit and constant encouragement of my thesis, research, and courses' studies. It is my most honor and pleasure to meet Professor Azzedine Boukerche and to have him as my supervisor during my master study.

I would also like to extend thanks to the members of the PARADISE Research Laboratory in University of Ottawa for their generosity, particularly Dr. Zhenxia Zhang. I am grateful for his help to my research and thesis.

Contents

List of Tables

List of Figures

Glossary

Mobile Ad-hoc Network (MANET) A network of mobile devices connected by wireless links.

Vehicular Ad-hoc Network (VANET) One type of MANET. A network of moving vehicles running on the roads connected by some wireless technologies; the transmission range is usually between 100 and 300 meters.

Intelligent Transportation Systems (ITS) Information and communication technologies are applied to vehicles or road side units to improve the safety, the transportation efficiency and some value-added services.

Vehicle-to-Vehicle (V2V) / Inter-Vehicle Communication (IVC) An important field in VANETs. Communication technology and protection of its security should be provided before deploying and applying VANETs.

Vehicle-to-Infrastructure (V2I) / Roadside-to-Vehicle Communication (RVC) Another important field similar to the V2V. This is the communication between vehicles and the units or infrastructures by the roads.

Greedy Forwarding Algorithm (GFA) One simple routing algorithm. Nodes forward the packet to a node which is geographically closest to the destination.

Global Positioning System (GPS) A space-based satellite navigation system that provides location and time information anytime and anywhere on or near the Earth [2].

Sybil attack One type of position-based attacks. One vehicle viciously pretends to be two or several vehicles at the same time.

Network Simulator 2 (NS2) Ns-2 is a discrete event simulator targeted at networking research. Ns-2 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. [4]

Radio Frequency (RF) A rate of oscillation in the range of about 3 kHz to 300 GHz, which corresponds to the frequency of radio waves, and the alternating currents which carry radio signals. [6]

Radio Signal Strength (RSS) The magnitude of the radio signal which can be used for the signal receiver to calculate the distance the signal has traveled.

Cooperative Location Verification (CLV) The location verification algorithm proposed by this thesis.

Secure Location Verification (SLV) A location verification algorithm proposed in [64].

Reputation Management System (RMS) The system can calculate and maintain trust values for every entity, like the credit for people in the human society.

Malicious/Misbehaving Nodes Those nodes trying to disturb the order of the network and compromising other nodes for some vicious purposes.

Cooperative/Benign Nodes Those nodes following the rules of the networks and trying to cooperate with other nodes.

Sequential Probability Ratio Test (SPRT) A specific sequential hypothesis test method in statistics.

Chapter 1

Introduction

1.1 Background of VANET

In recent years, mobile ad-hoc network (MANET) has been a very hot research area. MANET is a self-configuring network of mobile devices by wireless [3]. If such mobile devices are deployed in the vehicles, then all the vehicles running on the roads can build a network named as vehicular ad-hoc network (VANET). On the other hand, as more and more people can afford cars and a growing attention is given to the intelligent transportation systems (ITS), VANET becomes one of the most concerned area in MANET now and has been involved in many projects [5] [7] [1]. ITS includes wireless communications, computing, sensor and control and management strategies to improve the safety of people driving cars and inform drivers of the conditions of roads [12]. Obviously, the demand of communication is a big issue in ITS and researches on VANETs could meet such need.

MANETs are featured by change of communication environment (different from fixed networks), vulnerability of wireless links, large range of user mobility and so forth [11]. VANETs, as one type of MANET, own several different but significant features. Firstly, the topology of VANETs changes in a high rate because of the high velocity of vehicles. Furthermore, safety applications are time critical and depend on reliable position

information. The importance of location information will be discussed in the following sections. And another important characteristic is that vehicles usually are equipped with Global Positioning System (GPS) devices which are of great benefit to localization. In addition, it should be noted that in VANETs the power constraint can be neglected because of recharging batteries [42].

In VANETs, vehicle-to-vehicle (V2V) and vehicle-to-roadside (V2R) communication by wireless are the principal and basic technologies [27]. V2V communication is usually viewed as a critical element of VANETs and needs vast improvement. In [10], the authors simulated traffic on a section of I-880 in Hayward, CA and traffic data was collected on March 13, 1993. Blum et al., based on the simulation results, found some difficulties in handling the inter-vehicle communication (IVC). The first problem is that the IVC network topology changes too rapidly that to manage. Although the movement of vehicles is constrained within roadways, the network topology still changes very fast because of the high relative speed of vehicles. Secondly, frequent fragmentation poses a challenge for the IVC. The nodes are unable to reach other nodes in nearby regions. However, this can be improved by enlarging the radio communication range. The third challenge is that the short lifetime of connectivity between nodes impedes the routing in VANETs. The last challenge is that it's unlikely to add redundancy which can provides additional bandwidth and security features. This is due to the fact that, unlike other sorts of MANETs, VANETs causes severe limitation in time and function.

1.1.1 Applications in VANET

There are two main categories of application in VANETs [67]. The first one provides the basic applications to meet the fundamental demands including two aspects: safety and transport efficiency. Safety applications can inform drivers of the surrounding road condition and prevent the potential accidents [13]. For example, as shown in the Fig. 1.1, one big truck runs between the two vehicles, hindering them from noticing each other. However, if there is an application working on the vehicles, the two vehicles can sense

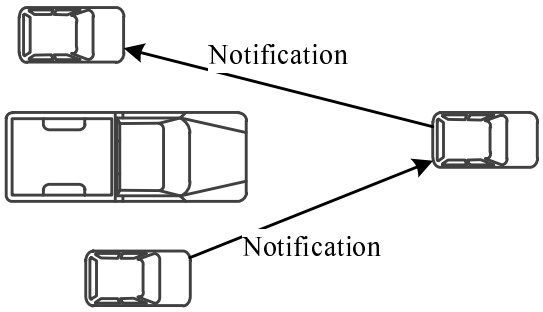

Figure 1.1: Improve Safety of Drivers in VANETs

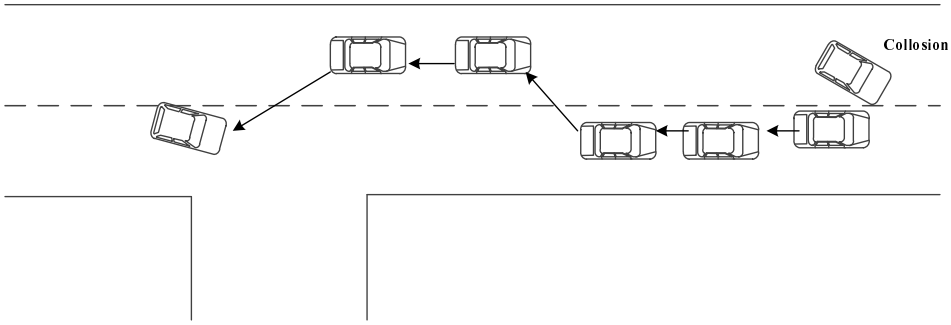

Figure 1.2: Improve Transportation Efficiency in VANETs

each other with the help of another vehicle, to avoid any potential danger. Certainly, there also exist other cases in which the view of other vehicles is obstructed by high buildings, trees, defects of road design and so forth. Additionally, transport efficiency applications can make suggestions for travel routes to the driver, to avoid traffic jams, traffic red lights and so on, for the purpose of saving time. Fig. 1.2 shows an example in which one vehicle successfully avoided a traffic jam caused by collision. The second field of application is to offer some value-added services, for example, entertainment.

1.1.2 Routing in VANET

There are many routing algorithms have been proposed in the literature. Li et al. in [37] summarizes that five categories of routing protocols have been researched. They

are position-based routing [40] [45] [31] [44], cluster-based routing [9] [62], broadcast routing [22] [65] [32] [26], geocast routing [14] [46] [47] and ad-hoc routing [53] [30]. In VANETs, vehicle movement is usually restricted in bidirectional movements along roads and streets, so the importance of geographical location is prominent. Some studies [41] [23] have supported the proposition that position-based routing protocol is a more promising routing strategy for VANETs by comparing the performance of topology-based routing against position-based routing strategies.

GPSR (Greedy Perimeter Stateless Routing) [31] is one of the best known position-based routing protocols in literature. GPSR has two modes. Usually, an intermediate node will forward the packet to the neighbor that is geographically closest to the destination, which is called greedy forwarding algorithm. If the intermediate node cannot find any neighbor who is closer to the destination than itself, then the packet will switch itself to perimeter mode at this intermediate node (N). In the perimeter mode, the nodes always forward packets by right-hand rule. The rule can be described as follows. Firstly the node will draw a virtual vector from itself to the destination node; then it will forward the packet to the first edge counterclockwise from the virtual vector. The edge here means a virtual direct link between two vehicles. If the packet still stays in the perimeter mode, the next forwarding node will continue to forward it to the first edge from the previous edge without crossing the initial vector. When the packet arrives at a node which is closer than the initial node (N), it will switch back to greedy forwarding mode. If it is forwarded back to N without switching to greedy forwarding mode, the packet will be dropped. Fig. 1.3 shows an example of how GSPR works. Source vehicle S needs to send a packet to D which is out of S's transmission range. Then the packet switch to the perimeter mode and forward the packet to A by counterclockwise search. A then forward it to B by another counterclockwise search. Because B is closer to D than S, the packet change back to greedy forwarding mode. Finally, it is forwarded to D successfully.

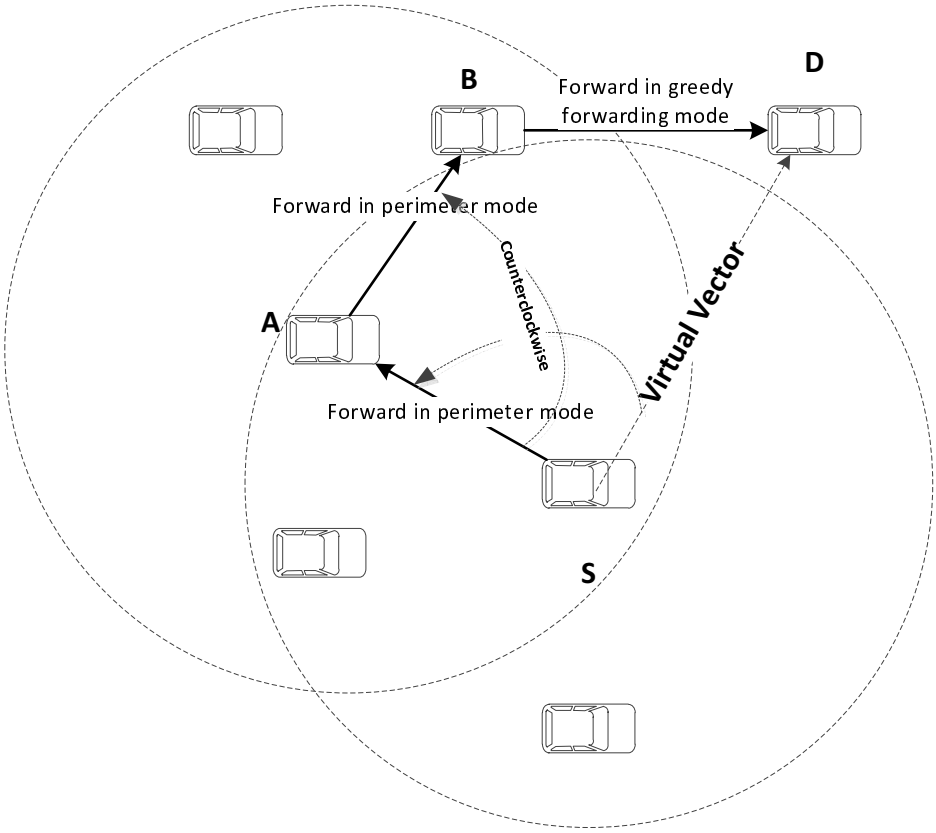

Figure 1.3: An example of GPSR

1.1.3 Security and Privacy

Security is very critical in VANETs. VANETs must satisfy several significant security issues before they can be deployed which has been discussed in many previous works [27] [67] [18] [38] [54] [58] [55].

There are three kinds of security problems having been researched. The most important one for VANETs is authentication. A vehicle should be able to verify the information provided by other vehicles to guarantee the correctness. Such information could be location, identity, collision report and so on. Besides, a vehicle should also detect whether a received message is modified by other vehicles after the source vehicle sends it. The second category is related to privacy. The network should guarantee that the identity

and other private information cannot be illegally retrieved. The third one is close to privacy which is named as liability. It means the network administrator (such as police) can check the identity and any other private information when they are necessary. Obviously, it's difficult to overcome the privacy issue meanwhile maintaining the accessibility to the private information for some security agencies.

The security and privacy are very significant in VANETs; therefore, there are numerous researches in this area and here are some solutions to prevent all kinds of security attacks [35] [70] [43] [15] [69] [56] [33] [39]. The solutions in [35] [70] are to prevent position-based security attacks, which is the focus of this thesis; hence they will be discussed in the section of Related Work. The other solutions aim to handle the rest security attacks by creating a traffic situation and broadcasting a traffic warning message [43], affecting the privacy of Vehicie-to-Infrastructure (V2I) communications [15], disrupting the networks by stopping to respond and drop packets [56], compromising the identity of a vehicle [33] and changing the identity of itself in order to prevent from being tracked during the attack [69] [39]. A briefly description of one of them will be given to help the understand of the security and privacy in VANETs.

In [43], Lo et al. proposed five rules to prevent the unreal traffic warning messages. In order to explain those rules, they assumed one warning message should include the following information: event ID, event type, the location where the event occurred, timestamp representing the time when the event took place, hops (the total number of hop counts for message forwarding) and velocity (the average velocity of the source vehicle). The first rule is dropping the duplicate message according to the event ID. The second one is checking the hops of the message whether to exceed the estimated maximum hops. The different types of events have various impact duration because one event should not exists in the networks forever; hence, the sum (T_{max}) of such impact duration time and people's reaction time will be the maximum time of the event's existence. Then the maximum hops of the event is calculated based on the T_{max}, the average velocity of the source vehicle and the transmission range for a vehicle antenna (R). The third rule

is that the message receive must check the location information based on the fact that the distance between the location where the message is sent and the current location of receiver must be less than the distance that the message has traveled. The fourth rule is checking the correctness of the timestamp based on the valid period of time for this type of message and the synchronization tolerance of the time. And the last rule is that the average velocity must be less than the highest speed limit in surrounding area. This is a simple case in the solutions to prevent security attacks, but it reflects many characters of such designs in VANETs.

1.2 Motivation

A lot of position-based routing protocols have been proposed to be used in VANETs [17]. In position-based routing algorithm, such as Greedy Forwarding Algorithm (GFA), vehicles can get their own location by using GPS and broadcast the location information regularly. Then, every vehicle can maintain a table to record its own neighbors. Specifically for GFA, every vehicle will forward packets to the neighbor closest to the destination among all the neighbors. Although GFA is one of the simplest routing methods, it's still obvious to find that position information is a very important parameter for position-based routing. For example, a vehicle may claim a false position in order to appear more optimal than other candidates to aggregate all data as a black hole [29]. In Fig. 1.4, source vehicle S should forward one packet to vehicle A which is nearest to destination vehicle D. However, malicious vehicle B claims its position in front of vehicle A, causing S to send the packet to B instead of A. Then, malicious vehicle B can modify the packet or discard it.

In another example, the routing algorithm GPCR [45], the vehicles prefer to send packets to the vehicles on a junction, compared to sending them across the junction. Hence junctions are the only places where actual routing decisions are made. However, if a malicious node running on the road always pretend to be on a closest junction to

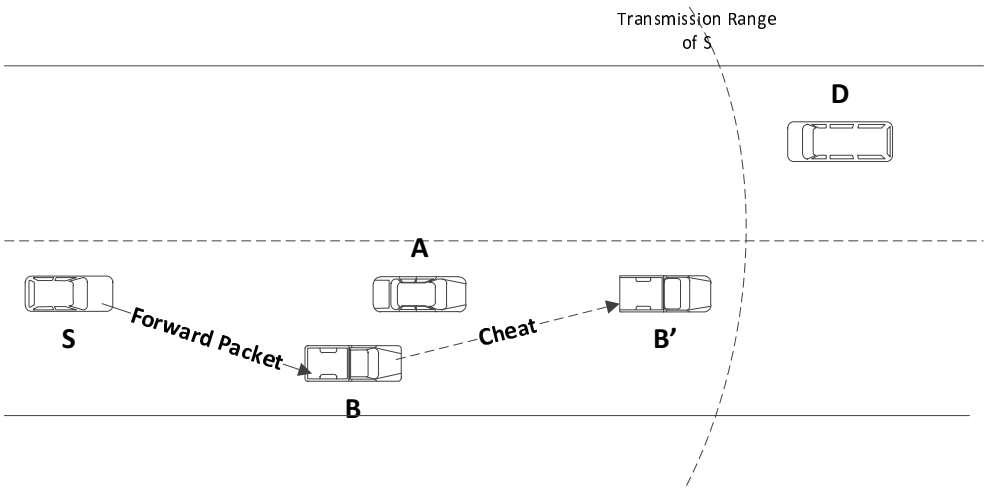

Figure 1.4: Example of Position-based Attack

itself, it can almost control a significant routing status.

Therefore, the malicious vehicles could receive more packets by pretending to be in the most optimal position based on the routing algorithm; they then can take more benefits from the private information in the packets. Hence security and verification of location information are very critical to position-based routing protocols, one of the most promising routing algorithms in VANETs. Before talking about the location verification algorithms, the types of position-based attacks and their impact should be discussed in order to propose some localization verification algorithms.

1.3 Problem Statement: Position-based Attacks

1.3.1 Classification of Position-based Attacks

To the best of the thesis author's knowledge, there are three categories of position-based attacks in the current literature. The first category is position cheating and false position disseminating [34]. In other words, a faking vehicle may change its own position and disseminate the false position to achieve some vicious objective. Thus, the geographic routing will be affected. The second category is the forging of positions which means

malicious nodes may modify other node's position packets, such as replaying bogus position packets, dropping urgent position packets and so on [61]. The last category is Sybil attack [51], also known as illusion attack. This is a well-known harmful attack whereby a vehicle claims to be several vehicles either at the same time or in succession. This thesis will mainly focus on the first category and the other two will be covered in our future work.

1.3.2 Impact of Position-based Attacks

This section will summarize the impact of position-based attacks on geographic routing. In the paper [36], Leinmüller et al. researched the influence of falsified position data on geographic ad-hoc routing; this reflected that the existence of malicious nodes can lead to a serious drop in the overall packet delivery ratio. They implemented position faking in the ns-2 simulator to study the impact of falsified position data. This study showed that falsified position information can impair the overall successful delivery of messages. The ratio of success is one of the most significant evaluation parameters in VANETs. As the paper presented, if 40 percent of the nodes are involved in position faking and still forwarding packets in the network, this could cause a 50 percent packet delivery ratio. If those position faking nodes drop packets instead of forwarding them, only about 10 percent of the packets could be delivered to the destination. Additionally, they demonstrated that a large network field size could lead to a higher probability of encountering a malicious node, compared to a small network field. Similarly, for a sparse network, the packet's delivery ratio decreases and leverages the effects of dropping. To summarize, we need location verification strategies to prevent the position-based attacks so that VANETs can reach a high packet delivery ratio.

1.4 Contributions

This thesis will mainly focus on the position-based attack which may affect the position-based routing of packets. I will resolve how one node can verify the correctness of the location claimed by other nodes. The main contributions of this thesis can be listed as follows:

- A RF-based location verification approach for VANETs will be proposed in this thesis to prevent position-spoofing attacks. A series of rules for vehicle selection are designed to help the verification process. The proposed approach guarantees a higher level of security than other similar approaches.
- A reputation management system is designed to maintain reliability for vehicles in the network. In addition, a solution will be provided to guarantee the security of the sparse network.
- Finally, the simulation results indicate that our approach can achieve a higher level of security than both GFA and SLV [64]. The effectiveness of the reputation management system will also be proved.

1.5 Outline of Thesis

The remainder of this paper is organized as follows. In the Chapter 2, I will summarize important location verification schemes in the literature. In Chapter 3, I will discuss our novel algorithm to verify the position information claimed by vehicles. In Chapter 4, our proposed reputation management system will be presented. Then Chapter 5 will provide an explanation for the comparison of our algorithm with others and reveal the simulation results. Finally, our paper will end by presentation of the conclusion and future work.

1.6 Summary

This chapter intends to provide basic knowledge in the area of my thesis. The first section is an overview of VANETs, including brief description of VANETs, comparison of VANETs to MANETs, features of VANETs and so forth. Then three significant aspects in VANETs are introduced: routing, applications and security. The second section, manifests the motivation of my research: to prevent the position-based attacks as the location information is quite critical in VANETs. And the third section discusses the classification and impact of the position-based attacks. The next section shows the contributions and then it is the organization of my thesis.

Chapter 2

Related Work

This chapter will summarize the literatures of the fields my research has involved. Besides, some personal viewpoints about the advantages and disadvantages of each method will be given.

2.1 Detection of Position-based Attack

In the current literature, location verification approaches usually are classified into two categories: infrastructure-based and infrastructure-less [28]. The design space has been discussed in the paper [34]. Usually, there are two types of approaches in the infrastructure-based category: dedicated measuring hardware required and no special hardware. As for the infrastructure-less category, there are autonomous verification and cooperative verification 2.1.

In the infrastructure-based category, most approaches use the road-side unit to help the verification or deploy some base station to manage data. However, there are two difficulties for such approaches to be implemented. The first one is that the infrastructure deployment and maintenance will increase the cost which definitely hinders the development of VANETs. In addition, such approaches rely on the infrastructures too much; hence, the number of infrastructures deployed will become the bottle neck in the

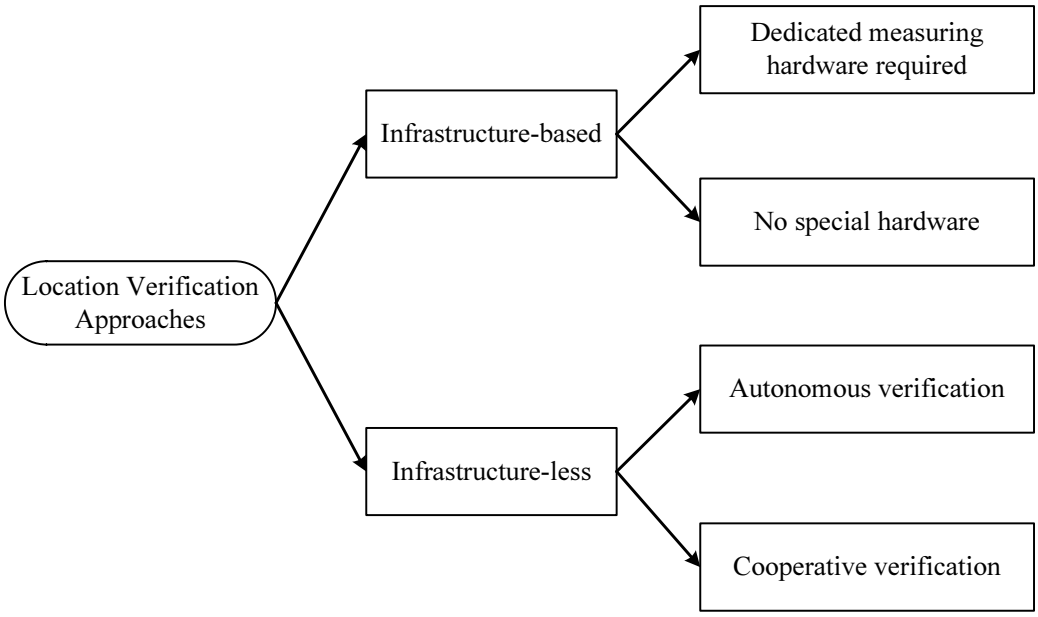

Figure 2.1: Location Verification Approaches Design Space

networks.

On the contrary, the infrastructure-less approaches are more suitable for VANETs. Most of them may make use of the radio signal strength, the signal's time of flight (ToF), the challenge-response procedures and so forth. After reviewing the literature related to the location verification approaches, it is found that the number of the infrastructure-less approaches is greater than the infrastructure-based approaches.

Now, several location verification schemes in the current literature will be summarized (see Table 2.1).

In [63], Sastry et al. proposed a classic secure location verification approach for sensor and wireless networks, namely Echo protocol. The remarkable character of this protocol is that it is extremely lightweight. The protocol is quite simple. We assume that there are two nodes (V and P). As definition of the Region of Acceptance (ROA) in the paper, the ROA of V is the area in which V is sure that it can correctly verify claims for P. If the node P declares it is in the ROA of V, then in order to verify it, V sends a

Table 2.1: Summary of Location Verification Proposed for VANET

Authors	**Features**	**Infrastructure**
Sastry et al. [63]	Extremely lightweight; In-region verification	No
Osama et al. [8]	Verify when the direct connection is not possible	No
Chen et al. [16]	Rely on digital signatures authorized by RSU	Yes
Leinmuller et al. [35]	Implement a series of sensors	No
Yan et al. [70]	Equip radar to every vehicle	No
Song et al. [64]	Ellipse-based detection based on TDoA	No
Ren et al. [60]	Equip two directional antennas to every vehicle	No

packet including a random number to P using RF. P has been supposed to reply an echo packet including the same number once it receives the challenge packet from V by using ultrasound. Such number can prevent P from replying before it receives the packet. V can predict and calculate how long it should take to hear the echo, namely, the sum of the time it takes to reach P by using RF, plus the supposed packet processing delay, plus the time it takes for the respond from P to V using ultrasound. Then if the measured elapsed time exceeds this expected time, V will reject the previous location claim from P.

In [8], Osama et al. proposed a cooperative multi-hop approach to verify a claimed location when direct communication is not possible because of the existence of an obstacle. In the algorithm, when one node V needs to verify node P's position but cannot connect P directly, then V will broadcast its request to find a node N which has a direct

communication with P. Certainly, one hop broadcast may not satisfy the demand because all the neighbors of V may not connect P directly. Hence, they also proposed a series of multiple-hop rules to find the objective node who has the direct connection with P. When such node N receives the request, it will estimate its distance from P according to radio signal strength (RSS) and reply such information to the last-hop node based on the routing chain of the request from V. Then every node receiving the response will compute its distance from P based on the response and determine whether the result is right or not by comparing it to the distance information already on its own table. If the distance information in the response is reliable, node V will get the response finally. Thus, node V detect node P successfully without direct connection. They also evaluated his algorithm by comparing to single-hop beacon and multiple-hop beacon and he showed the proposed scheme successfully improved the average awareness rate.

Chen et al. [16] proposed a robust method of Sybil attack detection in urban VANETs with limited infrastructures. The roadside units (RSU) can authorize digital signatures to vehicles in motion. Such infrastructure should broadcast the tamper-free digital signatures with time-stamp to vehicular nodes periodically. And every vehicle needs to record a set of signatures assigned by different infrastructures and the relevant time to form signature vector (infrastructure's number, time). In order to detect whether Sybil nodes exist, one node can collect the signature vectors of all its neighbors and comparing them based on the rule that the different real vehicles should have the differences in physical vehicles position and motion trajectories. Thus, the node can successfully detect whether Sybil nodes exist and classified all Sybil nodes fabricated by the same vicious node into a set. Although, the infrastructure does not need the complex centralized certification and revocation which are used in many other infrastructure-based methods, the method in the paper is still expensive to deploy the infrastructures which makes it difficult to be implemented.

In [35] [34], Tim Leinmüller et al. proposed to use a number of different independent sensors to recognize nodes cheating about their location. They basically used two

categories of sensors: the first one includes threshold-based sensors, map-based sensor and overhearing sensor. There are three types of threshold on which the threshold-based sensors are based: transmission range, maximum velocity and maximum density. For example, if one vehicle report its current location which exceeds the maximum range calculated by using its last reported location and maximum velocity of the road, then it will be recognized as a malicious vehicle. If vehicles can access to the street maps, the map-based sensors can detect whether one vehicle‘s claimed location is impossible, such as off the streets. As for the overhearing sensor, it can help other vehicles to verify the claimed location information. Such overhearing method, which is originated from [48], is worth researching and improving. Overhearing is a concept where nodes use the so-called promiscuous mode to capture the packets sent by nodes in transmission range but are addressed by other nodes. As the topology shown in the Fig. 2.2, we can learn two cases of how the overhearing works. First, the vehicle (V) needs to forward a packet and A' (the position where A pretends to be) is the best choice to forward the packet based on the routing algorithm. After forwarding, V overhears the next hop of the packet and find that A forwards the packet to B which is at an inferior position compared to A'. Thus, V can realize that A is not at the location reported by A (A'). In the second case, V overhears that C sends a packet to A. However, as the local position information shows, A should locate at A' where is outside of the transmission range of C. Therefore, V can detect A is cheating about its position. The second category of sensors are cooperative sensors. They can exchange neighbor tables with other so that nodes can share the information of one node's reliability. The other cooperative sensor is used to send position request which can be triggered when a node needs to verify one neighbor. Finally, they set the different weight values for every sensor which determine the different significance. Thus, by combining the results of all sensors, nodes can make a decision about whether trust one node or not.

In [70], Yan et al. proposed a GPS and radar integrated detection. He used an on-board radar as the virtual eye of a vehicle. Although the radar‘s transmission range is

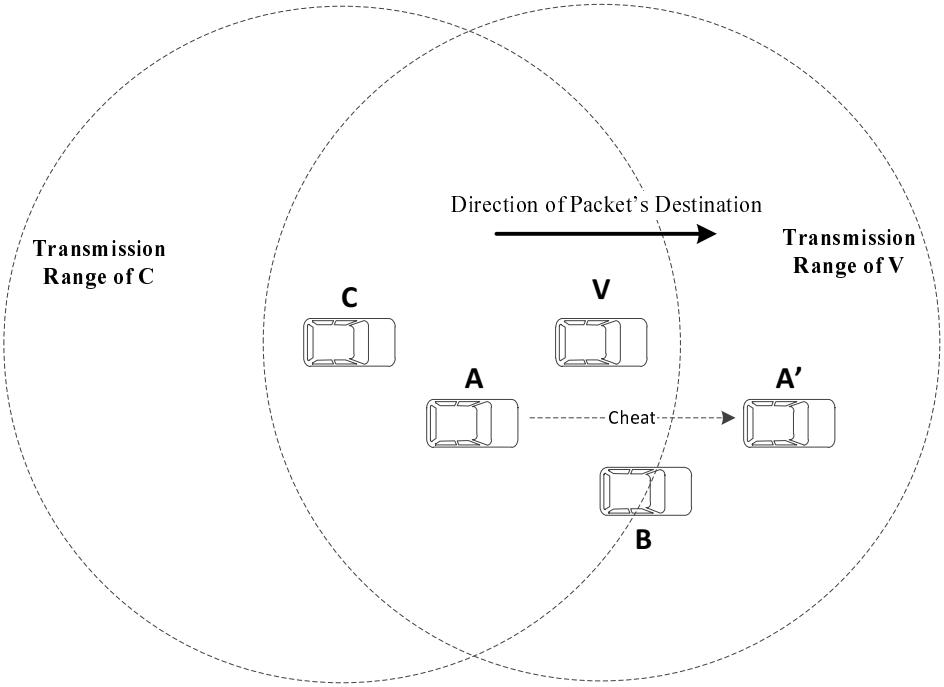

Figure 2.2: Overhearing

very limited, it has a much more precise detection than GPS. They take the tolerance of both GPS and radar-detection into consideration and combine the detecting result of GPS and radar to determine whether the claimed GPS location is reliable. The authors also extended such local security to global security. In the scenario, the network is organized by many preset position-based cells. In one cell, there is one cell leader and several cell routers to maintain the global security based on a series of proposed rules.

In the secure location verification method [64], Song et al. proposed a RF-based verification which needs to choose a common neighbor which locates within the transmission range of both Verifier and Prover to help the verification. Basically, when one vehicle Verifier (V) needs to verify a claimed position of another vehicle Prover (P), it will choose an appropriate neighbor (N) to verify the claimed position for vicious distance enlargement using ellipse computation and based on the time difference of arrival (TDoA). In specific, firstly V will broadcast a position request to P. N will start a timer as long as it

receives such request and P will broadcast its own position information once it receives the request. When N receives the broadcasting response from P, it will stop the timer. Such time recorded by the timer is called as TDoA. Then, N can to verify P's claimed position by using P's claimed position information and TDoA.

In the [60], Ren et al. proposed the relative position verification mechanism; this requires that every vehicle is equipped with two directional antennas. And then the vehicle can divide its neighbors into two groups. Then, nodes can exchange neighbors grouping information with other vehicles to detect the malicious nodes based on two detection rules. In the first step, nodes will generate two vectors: one records the nodes in front of it and the other maintains the vehicles behind it. After dividing neighbors into proper groups, each node will broadcast the group information periodically. According to all the broadcasting messages, one node can detect the malicious nodes by using the first detection rule: for a certain node i, j is not in the transmission range of i and the beacon message of node j is received from the f-antenna of i; this means j is in front of i; obviously, all the neighbors of j will be located in the front of i. Hence, one message showing one neighbor of j is located behind i will prove j is malicious. The second detect rule is that two neighbors' grouping information for each other should not conflict. For example, node i locally records node j is in front of itself, but the message from j shows it is behind i. This proves that j is malicious because i certainly know j's relative position by checking trough which antenna this message is received. However, this approach seems to be not practical because the directional antenna is hard to be accurate and easy to be affected by obstacles, such as buildings, trees and so on. Besides, deploying such antennas to every vehicles is very difficult to implement.

2.2 Reputation Management

After reviewing the literature of VANETs, the reputation management (trust management) for VANETs have been summarized (see Table 2.2). In the paper [71], Zhang

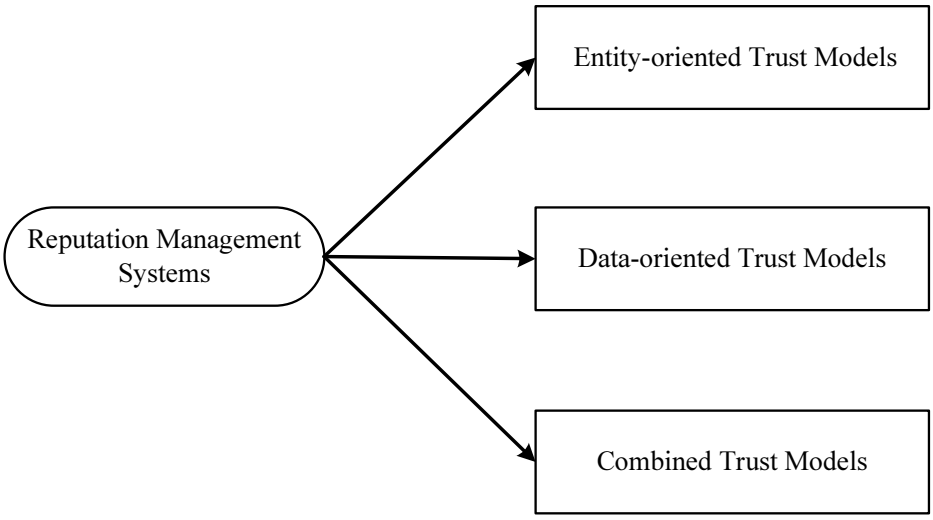

Figure 2.3: Categories of Reputation Management Systems

summarized three categories of trust models, entity-oriented trust models [24] [49], data-oriented trust models [57] [25], and combined trust models [21] [52] [72] (see Figure 2.3). The first category focus on the reputation of every peer in the network and maintain peer trust history over time. Data-oriented trust models mainly pay attention to the trustworthiness of data. For combined trust models, they combine the two kinds of trust models mentioned before in order to offer a more stable and reliable network. For example, a combined trust model may use an algorithm to verify every single message sent by peers to guarantee the trustworthiness of data. Then the reputation of each peer can be generated by counting the ratio of the number of right messages to the number of wrong messages. Similarly, the model also may use the reputation of one peer to determine the rightness of its messages. Obviously, combined trust model is a promising strategy for VANETs. The scheme proposed in this thesis includes a reputation management system to manage the reliability record in the whole network. Technically, this proposed model belongs to the combined trust models.

In the paper [59], Refaei et al. not only proposed a reputation system for ad-hoc networks, but also summarized three main categories of functions for reputation man-

Table 2.2: Summary of Reputation Management Systems for VANET

Authors	**Features**	**Category**
Refaei et al. [59]	Adaptive reputation management based on time slot for Ad-hoc networks	Entity-oriented
Ding et al. [19] [20]	Calculate trust value for a traffic event based on the type of the vehicle and its time-stamp	Combined
Gerlach [24]	Apply a sociological trust model to vehicular networks and identify four types of trust	Entity-oriented
Raya et al. [57]	Establish the trust of data rather than the vehicles	Data-oriented
Dotzer et al. [21]	Propose the Opinion Piggybacking to manage the trustworthiness of one message	Combined

agement which could be convenient for researchers afterwards. Such three categories of functions are, respectively, behavior evaluation, behavior detection and reaction 2.4. Behavior evaluation is used to evaluate each node's behavior and assigns each one a score. In this part, I usually select a evaluation metric as the criterion to judge nodes. The second one, behavior detection uses the nodes' scores assigned by the behavior evaluation function to distinguish malicious nodes from all the nodes in the network. For example, we can set a threshold for the score and every node whose score is below the threshold will be considered as malicious nodes. The last category is reaction which takes action against nodes according to their behavior. In other words, it should punish the misbehaving nodes and reward the cooperative nodes. For example, the reaction function can

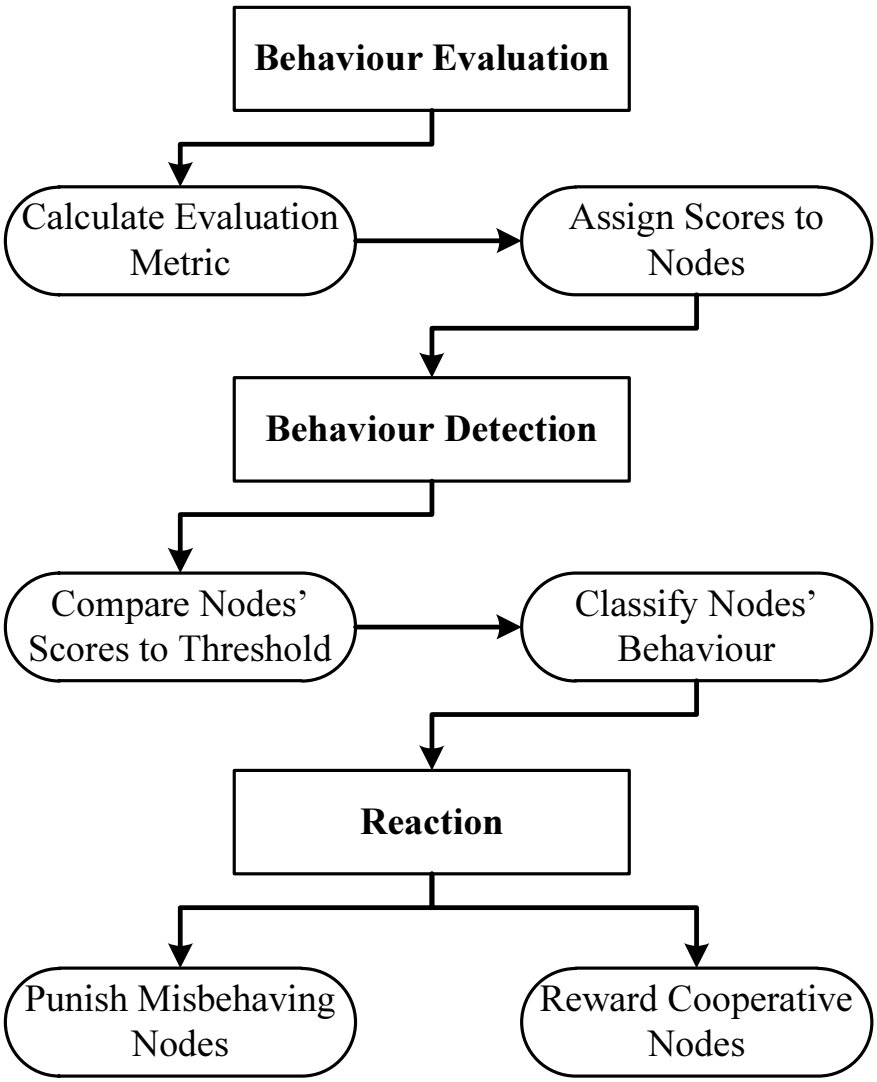

Figure 2.4: Three Categories of Function in Reputation Management System

be used to notify other nodes about the malicious nodes.

Now, I will give a brief review of the reputation management system the authors proposed. They proposed a novel idea about reputation management system which is adaptive according to observed volume of traffic. It evaluates nodes' behavior based on time slot whose duration will change as the volume of traffic changes. They selected the ratio of the number of packets forwarded by a node to the number of packets routed

through it as the evaluation metric. Thus, if the network is very active, then the duration of one slot will be short. On the contrary, if the nodes are sparse, then the duration of one time slot should be longer in order to collect enough data to evaluate the behavior. In the behavior detection functions, the authors use the sequential probability ratio test to analyze the scores of nodes to make a decision about their behaviors. Such process is also adaptive, for example, the number of observations needed to make an decision that one node is malicious will be small if it's possibility of dropping packet is low. It does make sense because a node is much more possible to drop a packet viciously in the case of a low possibility of dropping during previous time. Finally the author did the simulation in both static network and dynamic network to prove the validity of their adaptive reputation management system.

In the papers [19] [20], Ding et al. used a reputation system for VANETs to calculate a value for every traffic event, such as traffic accident. Vehicles are classified into three categories when an event happens: event reporter, event observer and event participant. A vehicle is called event reporter if it can perceive incident by equipped sensors. Vehicles within one hop with event reporter are event observers because they can receive the event message directly from event reporter. Then other vehicles locating beyond one hop with the event reporter are event participant. They can receive and forward the event message but cannot verify the behavior of event reporter. Vehicles will record every event by unique event identity, event type, time-stamp when traffic event occurred, reputation value of this event and so forth. And the author set global trust aggregation weight for every event by considering the last hop of the event (obviously event reporter is most reliable) and time-stamp (the newer the event is, the higher the weight will be). Finally, vehicles use a formula to calculate global reputation for events. In the paper, the authors did not consider the reputation of vehicles which makes it to be hardly practical.

In the paper [24], Gerlach proposed a sociological trust model and identified four forms of trust situational trust and dispositional trust. Additionally, the author applied such trust model to vehicular environments. Four major attributes that could form a

trust relation were summarized in vehicular networks: raw sensor information, higher level sensor information, services and attributes (such as being a vehicle, being a police car, treating information confidential, and); and based on these, trust tags including trust and confidence values in the interval [0, 1] are used to represent trust. However, the author did not provide any specific information about how to combine the different types of trust to form a trust value.

Raya et al. [57] proposed a framework for data-centric trust establishment. This may be more appropriate for the VANETs because the topology changes too fast to establish a very effective trust model for maintaining the trustworthiness of the vehicles. In their model, they set a default trustworthiness for every node based on the attributes related to the designated type of the node. And they then define the trustworthiness function according to the type of the reporting node and the task. The trust value certainly should not be static; therefore, a security status function and dynamic trust metric functions are designed to take the change of nodes' attributes into consideration. Location and time are two important attributes because it is obvious that the closer the reporter is to the location of an event, the more accurate the information reported by the reporter is. Similarly, the more recent to the event occurrence time a report is generated, the more accurate the report will be. By take into account various trust metrics, the model generates trust levels for data reports. It then combines them to make a decision by using decision logic. In the paper, the authors compared three decision logics, namely weighted voting, Bayesian inference and Dempster-Shafer Theory and they found Bayesian inference and Dempster-Shafer Theory are the most promising approaches to evidence evaluation.

In the paper [21], Dotzer et al. suggested a distributed reputation model that utilize the Opinion Piggybacking which means that each forwarding node of one message appends its own opinion about the trustworthiness of the data. They provided an algorithm to calculate and generate the opinions for forwarding nodes based on reputation level, previous nodes' opinion about the data and so on. Firstly, the forwarding node com-

bines all previous partial opinions with the reputation values of the relevant nodes who appended the partial opinions. If there is not the relevant reputation value of one node, the reputation value will be calculated based on indirect or direct experience. However, authors did not provide sufficient and complete details about the approach. In addition, there is a hidden trouble in the approach that the previous nodes' opinions will have much more significant influence on the data than the later nodes.

2.3 Summary

This chapter reviews the literatures of location verification approaches and reputation management system. As for the location verification approaches, seven approaches have been briefly described; six of them are infrastructure-less and one of them is infrastructure-based. There are two reasons: the first one is that researches on infrastructure-less approaches are more than the other one in the literature; the second one is that the protocol proposed in this thesis is also infrastructure-less because such approaches are more promising, easier and faster to be applied in VANETs than the infrastructure-based schemes.

As for the reputation management systems, five approaches in the literature are discussed. As far as this thesis author is concerned, the combined trust model is worthier of more studies because both entity-oriented model and data-oriented model are slightly partial and it will bring more benefits if they can be integrated effectively.

Chapter 3

Location Verification Approach

In this section, a location verification protocol called Cooperative Location Verification (CLV) will be introduced. We will illustrate how our protocol works. In addition, we also propose a reputation management system to maintain the reliability record of vehicles. Finally, we discuss how to deal with the sparse network.

3.1 Cooperative Location Verification

The aim of the protocol is to verify a vehicle claimed location by choosing one vehicle to help with the verification. The proposed protocol is based on the following assumptions:

Assumption 1: Each vehicle is able to determine its own position by using existing technologies such as GPS. GPS data normally changes in the range of 10m [66].

Assumption 2: Each vehicle is capable of communicating using radio frequency (RF) and counting time.

Assumption 3: Every vehicle will broadcast its own position information regularly and maintain a table recording that of all its neighbors. This is the basic requirement of geographic-based routing algorithms.

Assumption 4: Communication channels between vehicles are secure.

Assumption 5: Each vehicle has its own unique identity number.

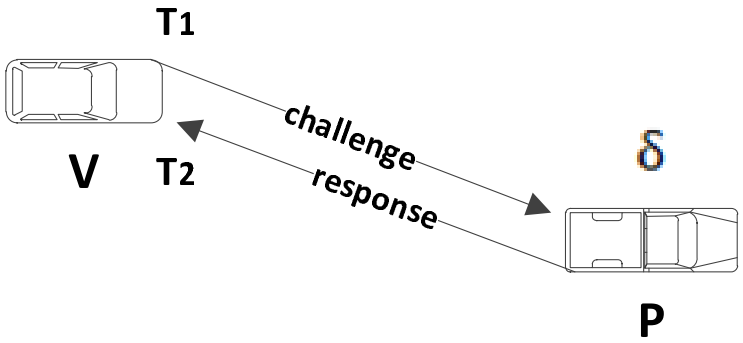

Figure 3.1: Challenge-Response Procedure

Assumption 6: Energy consumption and computation resources are not a major concern in VANET.

The proposed protocol is mainly based on one fact: the RF-based distance bounding technique can travel the minimum distance between Verifier V and Prover P. Such technique has been used in [63] and can be described as follows (Fig. 3.1): if V needs to calculate the distance between itself and P, V can start a timer at T_1 and send a challenge message which contains a random number (N) to P. P has to reply with its position information and N, immediately, as it receives the challenge. When V gets the response from P, it will stop the timer at T_2 and calculate the distance based on the time of flight (ToF) of such challenge-response messages and the speed-of-light c (transmission speed of RF signal). Because it is impossible for P to know when V will send a challenge message in advance, P cannot reduce the distance between itself and V. Furthermore, P can only falsify its location by viciously delaying the response message. Certainly, it is necessary to consider a signal processing delay δ for P. Therefore, ToF can be calculated by formula (3.1).

$$ToF = T_2 - T_1 \tag{3.1}$$

Our approach, which can detect such malicious behavior and thus guarantee the security in VANETs, is described as the following three steps that draw upon the notation

Table 3.1: Notations used in the algorithm

Variable	**Description**
V	Verifier Vehicle
P	Prover Vehicle: the objective of verification
C	Cooperator Vehicle
T	Transmitter Vehicle
NT	Neighbor Table
P'	Position (GPS coordinate) of P claimed by itself
N	An unpredictable random number
N'	Number replied by P
R_C	Verification result from C
d_{AB}	The distance between coordinate A and B
c	The speed of light
δ	The processing delay of P

in Table 3.1:

(1) Selecting Cooperator and Transmitter: When V needs to forward a packet to a destination, it will select one of its neighbors as the next-hop according to the routing algorithm. We call this next-hop Prover P. Then, V will send a request to P to acquire its neighbor table which maintains all neighbors of P and their location information. After that, V can select one vehicle as the Transmitter in the forwarding of messages between V and C according to the neighboring tables of both P and itself. Subsequently, V requires the table of T's neighbor and chooses the best Cooperator C based on the three neighbor tables of V, P and T (rules of selection will be discussed later). Cooperator C, which is located outside of the transmission range of V but within the transmission range of P, is used to help the verification of P.

(2) Verification by V: V and C will then execute a challenge-response procedure with P, respectively. At the beginning, V sends a challenge message to P. When P gets this message, it is required to broadcast its GPS coordinate (P') immediately. When V gets this message, it can detect whether P has tried to reduce the distance between V and P based on ToF_{VP} and P's claimed coordinate.

(3) Verification by C: once C gets this message, it will send its own challenge message to calculate the ToF_{CP} between C and P. C can then detect whether P has tried to enlarge the distance between V and P based on the ToF_{CP} and P's claimed position. The specific process will be discussed later.

Because the RF signal travels at the speed-of-light, the time of such a procedure is very short; therefore the distance P has moved during this procedure can be ignored. Hence, we consider that P is in the same position when it receives the challenge from V and C respectively.

3.2 Verifier's Verification

Fig. 3.2 shows an example of distance enlargement attack. After the challenge-response procedure between V and P, V can calculate the distance r_V between V and P according to ToF_{VP} (see (3.2)); based on the claimed coordinate P' ($x_{P'}$,$y_{P'}$) and its own coordinate (x_V,y_V), it can also calculate the distance $r_{VP'}$ between V and P' by using equation (3.3).

$$r_V = \frac{ToF_{VP} - \delta}{2} \times c \tag{3.2}$$

$$r_{VP'} = \sqrt{(x_V - x_{P'})^2 + (y_V - y_{P'})^2} \tag{3.3}$$

If the following inequality (3.4) is then satisfied, V will trust P; otherwise, V will recognize P as a malicious node.

$$|r_V - r_{VP'}| \leq t \tag{3.4}$$

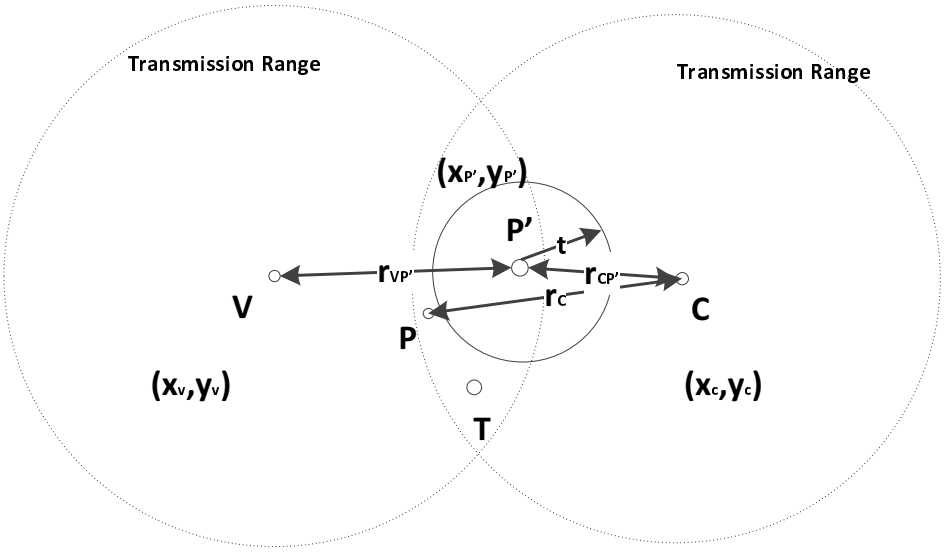

Figure 3.2: Verify P's Claimed Position

where $t = \delta c$. This procedure can detect the malicious node trying to reduce the distance (see Algorithm 1).

3.3 Cooperator's Verification

C can help V detect the distance enlargement attack by a challenge-response procedure between C and P. Similarly, C will calculate r_C and $r_{CP'}$. C can then determine whether to trust P according to the inequality (3.7). Such procedure can be seen in 2

$$r_C = \frac{ToF_{CP} - \delta}{2} \times c \tag{3.5}$$

$$r_{CP'} = \sqrt{(x_C - x_{P'})^2 + (y_C - y_{P'})^2} \tag{3.6}$$

$$|r_C - r_{CP'}| \leq t \tag{3.7}$$

Algorithm 1 Cooperative Location Verification: Verifier

Require: ID of P, C and T.

1: $V \rightarrow T : NeedHelp$
2: **if** get confirmation from C **then**
3: $T_1 \leftarrow time()$
4: $V \rightarrow P : N$
5: $P \rightarrow broadcast : N', P'$
6: $T_2 \leftarrow time()$ when V receives broadcast from P
7: **if** $N \neq N'$ **then**
8: **return** false
9: **end if**
10: **if** $T_2 - T_1 > \frac{2 \times d_{VP'}}{c} + \delta$ **then**
11: **return** false
12: **end if**
13: **end if**
14: Wait for R_C from C
15: **if** $R_C =$ true **then**
16: **return** true
17: **else**
18: **return** false
19: **end if**

3.4 Selection of the Best Cooperator and Transmitter

The selection of the Transmitter (T), which is used to exchange information between Verifier V and Cooperator C, is simple: a common neighbor of both V and C is chosen. However, T is selected before C, therefore, we choose one of V's neighbors as T which is nearest to P. This guarantees that we will have a relatively large range to select C.

Now, we will describe how to find the best C. In Fig. 3.3, d denotes the distance P has falsified; r_C represents the distance estimated by C's challenge-response procedure; $r_{CP'}$ refers to the distance between P's claimed position and C. Obviously, in order to prevent

Algorithm 2 Cooperative Location Verification: Cooperator

Require: ID of Prover (P), Cooperator (C) and Transmitter (T).

if get NeedHelp from T **then**
    $C \rightarrow T : confirmation$
end if
Wait for broadcast from P
if get broadcast (containing P') from P **then**
    $T_1 \leftarrow time()$
    $C \rightarrow P : N$
    $P \rightarrow C : N'$
    $T_2 \leftarrow time()$ when C receives N' from P
    if $N \neq N'$ **then**
        $C \rightarrow T : R_C = false$
        return false
    end if
    if $T_2 - T_1 > \frac{2 \times d_{CP'}}{c} + \delta$ **then**
        $C \rightarrow T : R_C = false$
        return false
    end if
end if
$C \rightarrow T : R_C = true$
return true

P's enlargement attack to V, C should be located on the different side of P than V (P is located between V and C); therefore, the value of θ is above 90° in Fig. 3.3. According to (3.7) which determines whether to trust P or not, C should be a node which can maximize the value of $|r_C - r_{CP'}|$. Now, we will discuss how to select C to maximize the value of $|r_C - r_{CP'}|$ with a certain cheat distance (d). Firstly, given a certain value of $r_{CP'}$, based on the law of cosines we have:

$$r_C = \sqrt{d^2 + {r_{CP'}}^2 - 2 \times d \times r_{CP'} \times \cos\theta} \tag{3.8}$$

Then, r_C taking derivative to $\cos\theta$ is

$$\frac{\partial r_C}{\partial \cos\theta} = \frac{-d \times r_{CP'}}{\sqrt{d^2 + {r_{CP'}}^2 - 2 \times d \times r_{CP'} \times \cos\theta}} < 0 \tag{3.9}$$

According to the property of cosines, the following equation is true:

$$\frac{\partial \cos\theta}{\partial \theta} < 0, \quad where\, 0 < \theta < 180^\circ \tag{3.10}$$

Because $\theta > 90^\circ$, we have:

$$\frac{\partial |r_C - r_{CP'}|}{\partial r_C} > 0, \quad where\, 90^\circ < \theta < 180^\circ \tag{3.11}$$

Therefore, based on (3.9), (3.10) and (3.11) we have:

$$\frac{\partial |r_C - r_{CP'}|}{\partial \theta} > 0, \quad when\, 90^\circ < \theta < 180^\circ \tag{3.12}$$

which means the greater the value of θ is, the larger the value of $|r_C - r_{CP'}|$ will be. Hence, the closer to the X-axis C is, the larger the value of $|r_C - r_{CP'}|$ will be. Secondly, given a certain value (z) of C's Y-axis coordinate, we can determine that the more distant C is to P', the larger the value of θ will be. Therefore, when the distance from C to the line formed by V and P' is a certain value (z), then the more distant C is to P', the larger the value of $|r_C - r_{CP'}|$ will be. Based on the analysis mentioned previously, we summarize the following rules for Verifier V to choose a cooperator C to verify P's claimed position.

(1) The first rule of selecting C is that such selection can locate P between V and C.

(2) C should be a neighbor of both P and T but not a neighbor of V. In other words, C should be within the transmission range of P and T but out of the transmission range of V.

(3) If there are multiple candidates satisfying rule (1) and (2), then V will choose one whose distance to the line, formed by V and P's claimed position, is the smaller than a predefined threshold. This rule is to guarantee that V, P and C are almost in one line.

(4) If there are multiple candidates satisfying the above rules, V will choose the one which is the most distant from V.

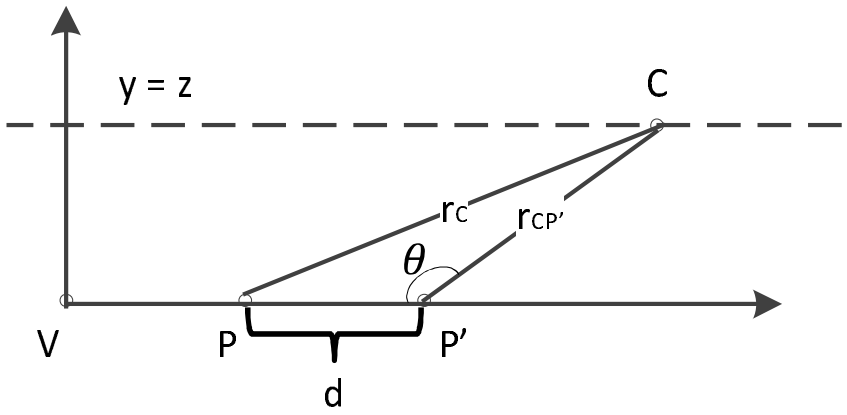

Figure 3.3: Selection of Best Cooperator and Transmitter

The Algorithm 3 and Algorithm 4 also show how to select the Transmitter and Cooperator. Such precess is based on the assumption that there is none information saved locally; hence the algorithm needs retrieve the neighbor table of other nodes by sending requests. However, in the real network which is using geographic routing algorithm, we do not need to require NT particularly because the routing algorithm will organize the nodes to broadcast location information periodically. Besides, the algorithm returns false means that it fails to select the Cooperator or Transmitter to verify P's location and the routing algorithm should select another node as the next-hop. And the algorithm returns true when it obtain the Cooperator and Transmitter successfully; then V will execute CLV to verify the P's location information.

3.5 An Example of Applying CLV

Fig. 3.4 shows a simple case to illustrate how our algorithm works. Vehicle S needs to send a packet to vehicle D. If the network applies the Greedy Forwarding Algorithm (GFA) which is required to select the node closest to the destination as the next hop, then S will choose vehicle B as the best potential next-hop to arrive at the destination (in this way B cheats its position as B'). Due to the application of our CLV algorithm,

Algorithm 3 Select the Best Transmitter

Require: ID of Prover (P). This is the next-hop selected by the routing algorithm

1: $V \rightarrow P$: *RequireNT*
2: Wait for NT from P and begin a Timer
3: **if** the request expires **then**
4: **return** false
5: **end if**
6: **if** get NT from P before the request expires **then**
7: V checks the common neighbors of both P and itself
8: **if** do not exist any common neighbor **then**
9: **return** false
10: **end if**
11: **if** exist one common neighbor **then**
12: Select it as the Transmitter
13: **end if**
14: **if** exist more than one common neighbor **then**
15: Select the node which is closest to P as the Transmitter
16: **end if**
17: **end if**
18: $V \rightarrow T$: *RequireNT*
19: Wait for NT from T and begin a Timer
20: **if** the request expires **then**
21: **return** false
22: **end if**
23: **if** get NT from T before the request expires **then**
24: Execute the algorithm of selecting the Cooperator
25: **end if**
26: **return** true

S selects an appropriate Cooperator to verify B's position and processes a challenge-response procedure with B before trusting this potential next-hop B. In keeping with the rules of CLV, vehicle T will be selected as the Transmitter and vehicle C as the Cooperator. When B receives the challenge from S, it viciously delays the broadcasting of its position B', so that it can successfully falsify its location. In this case, S cannot

Algorithm 4 Select the Best Cooperator

Require: ID of Prover (P) and Transmitter (T). NT of P and T

V checks the set (S) of the nodes which are neighbors of T but not neighbors of V; and P locates between such nodes and V
if S is empty **then**
    return false
end if
if S has one element **then**
    Select it as the Cooperator
end if
if S has more than one elements **then**
    Draw a virtual line (L) between V and P's claimed position (P') and calculate the distance to L for every node in S. Save such distances in the set (S_d)
    Check: $d < d_t$ for every $d \epsilon S_d$. d_t is the threshold to select an appropriate C
    if there is not such element in S_d **then**
        return false
    end if
    if there is such one element in S_d **then**
        Select the relevant node as the Cooperator
    end if
    if there are more than one such elements in S_d **then**
        Select the node which is the most distant from V as the Cooperator
    end if
end if
return true

detect B's falsehood by itself. Hence, S will wait for the result of verification from C. Then, Cooperator C will begin a challenge-response procedure with B once it receives the broadcast information (the claimed position) from B. C will estimate the ToF (t') based on its own position and the claimed position (B'). However, the minimum ToF of this challenge-response procedure is t (B will reply immediately once it receives a challenge from C), because B cannot know when C will send the challenge in advance. Thus C can successfully detect if B is cheating by comparing t to t' and sending its result to S

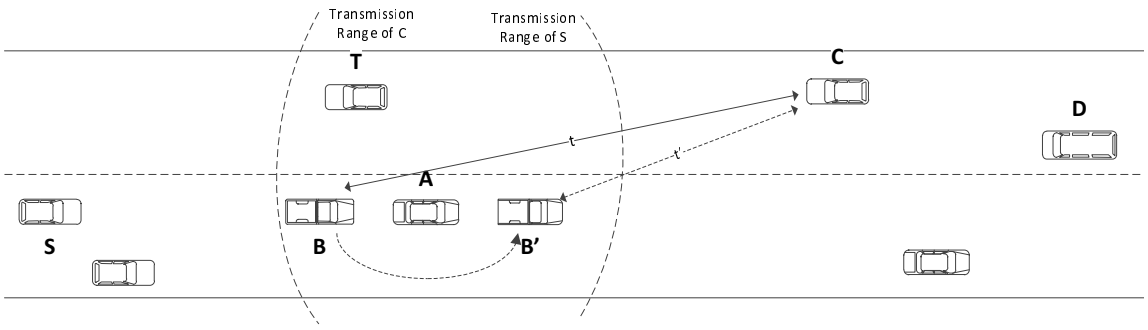

Figure 3.4: A Case of Applying CLV

through T. Finally, S will add B to its malicious nodes list and find another potential next-hop so that position-spoofing attacks can be avoided.

In addition, the reason of using ToF instead of the radio signal strength (RSS) will be given because the radio signal strength (RSS) also can be used to estimate the distance between two vehicles. If the RSS is used in the CLV, then the malicious nodes may increase or decrease the initial strength of the signal. In this case, the CLV will be useless. Hence the RSS is not suitable for the CLV.

3.6 Security Analysis

The proposed protocol relies on timing: the amount of time it takes to get a response from the Prover bounds how far the Prover can be from the Verifier. This section will show that it is impossible for a malicious vehicle falsifying its position to convince the Verifier that it is benign.

In the protocol, when the Prover receives the challenge carrying a random number (N) from the Verifier, it can viciously claim it's in the position P' to reduce or enlarge the real distance between itself and the Verifier (d_{VP}). In order to confirm that the Prover is at P', the Verifier must verify whether the response, which includes the outgoing number N, is received within:

$$t_{max} = \frac{2 \times d_{VP'}}{c} + \delta \tag{3.13}$$

where $d_{VP'}$ is the distance from the Verifier to the claimed location, c is the speed of radio propagation (approximately the speed of light) and δ is the Prover's processing delay.

Because we allow a maximum processing delay δ, our protocol only detect the difference between the real location and the claimed location is greater than $\delta \times c$. Thus, if the Prover viciously reduce the distance between itself and the Verifier, then we have

$$d_{VP'} < d_{VP} - \delta \times c \tag{3.14}$$

The attacker has only two choices: either make a guess concerning some of N's information, or learn the entire value of N from V. In the former case, we can choose N from a set of sufficient sizes to make the successful probability of such attack extremely small. In the latter case, we denote t_{finish} as the time at which the attacker finishes sending its response. Because the last bit of N will need d_{VP}/c and the attacker cannot finish transmitting its response before it has received the entire N, we have

$$t_{finish} > \frac{d_{VP}}{c} \tag{3.15}$$

Thus, based on (3.14)(3.15), the minimum time for the attacker to receive the entire N and get a response to V is

$$\begin{aligned} t_{min} &= t_{finish} + \frac{d_{VP}}{c} \\ &> \frac{d_{VP}}{c} + \frac{d_{VP}}{c} = \frac{2 \times d_{VP}}{c} \\ &> \frac{2 \times (d_{VP'} + \delta \times c)}{c} \\ &= \frac{2 \times d_{VP'}}{c} + \delta = t_{max} \end{aligned} \tag{3.16}$$

Hence, the Prover cannot reduce the distance between itself and the Verifier without being detected by the Verifier.

Similarly, if the attacker enlarges the real distance between itself and the Verifier, then we have

$$d_{VP'} > d_{VP} + \delta \times c \tag{3.17}$$

In this case, Verifier cannot successfully detect such attack, however, we selected a Cooperator to help the verification in our protocol. According to the rules we have mentioned previously, the Cooperator will approximately locate in one line with the Verifier and the Prover and the Prover is between the Verifier. Moreover, the Cooperator; therefore, we have

$$d_{CP'} \approx d_{VC} - d_{VP'} \tag{3.18}$$

Based on (3.17)(3.18), we have

$$d_{CP'} < d_{VC} - (d_{VP} + \delta \times c) \approx d_{CP} - \delta \times c \tag{3.19}$$

Similarly, in the challenge-response procedure between the Cooperator and the Prover, we have

$$t'_{finish} > \frac{d_{CP}}{c} \tag{3.20}$$

where t'_{finish} represents the time at which the attacker finishes sending its response to the Cooperator. Then, according to (3.19)(3.20), the minimum time for the attacker to receive the entire N and get a response to C is

$$\begin{aligned} t'_{min} &= t'_{finish} + \frac{d_{CP}}{c} \\ &> \frac{d_{CP}}{c} + \frac{d_{CP}}{c} = \frac{2 \times d_{CP}}{c} \\ &> \frac{2 \times (d_{CP'} + \delta \times c)}{c} \\ &= \frac{2 \times d_{CP'}}{c} + \delta = t'_{max} \end{aligned} \tag{3.21}$$

where t'_{max} is the time estimated by the Cooperator, in which it should obtain the response from the Prover, otherwise the Cooperator will consider the Prover as an attacker.

Consequently, any Prover trying to viciously reduce or enlarge the real distance between Prover and Verifier can be detected by our protocol.

3.7 Summary

This chapter discusses the Cooperative Location Verification (CLV) and illustrates the design of it. At the beginning, the RF-based challenge-response process is introduced, which is the basis of the CLV. Then the overall process of the CLV is presented: selecting the best Cooperator and Transmitter, verification executed by the Verifier and verification carried out by the Cooperator. Finally, the verification results from the Verifier and the Cooperator are integrated to judge whether to trust the claimed location of the Prover. In this process, the Transmitter is used to forwarding packets between V and C.

The next section shows a case of the CLV which helps a further understand of the CLV. In the following section, the security analysis of the CLV is shown to prove the validness of my design theoretically and mathematically.

Chapter 4

Reputation Management

We propose a new reputation management system (RMS) to estimate the reliability level for every vehicle based on the reputation value; and the RMS will divide the history into many slots based on time. The sequential probability ratio test (SPRT) will be used to estimate the reputation value based on the previous history records when the reputation value is not enough in the current time slot.

4.1 RMS based on Reputation Value

In our system, every vehicle maintains the Reputation Value (RV) of other vehicles, ranging from 0 to 1; other vehicles are divided into the following three categories: Trust, Unclear, Distrust (as shown in Fig. 4.1). Initially, all vehicles are placed in the Unclear category and their RVs are 0.5. Once a vehicle applies CLV to another vehicle, it will adjust this target vehicle's Reliable Category by using the rules (see Fig. 4.1). For example, if a vehicle in Unclear or Trust is verified as untrustworthy by local verification (CLV) and its RV is below 0.2, it will then be classified in the Distrust Category by the local vehicle; if a vehicle is in this category, it cannot return to the Trust category directly. Vehicles cannot move from the Distrust to Trust Category directly which can guarantee a higher level of security.

When a vehicle needs to forward packets to a neighbor, it will send them without

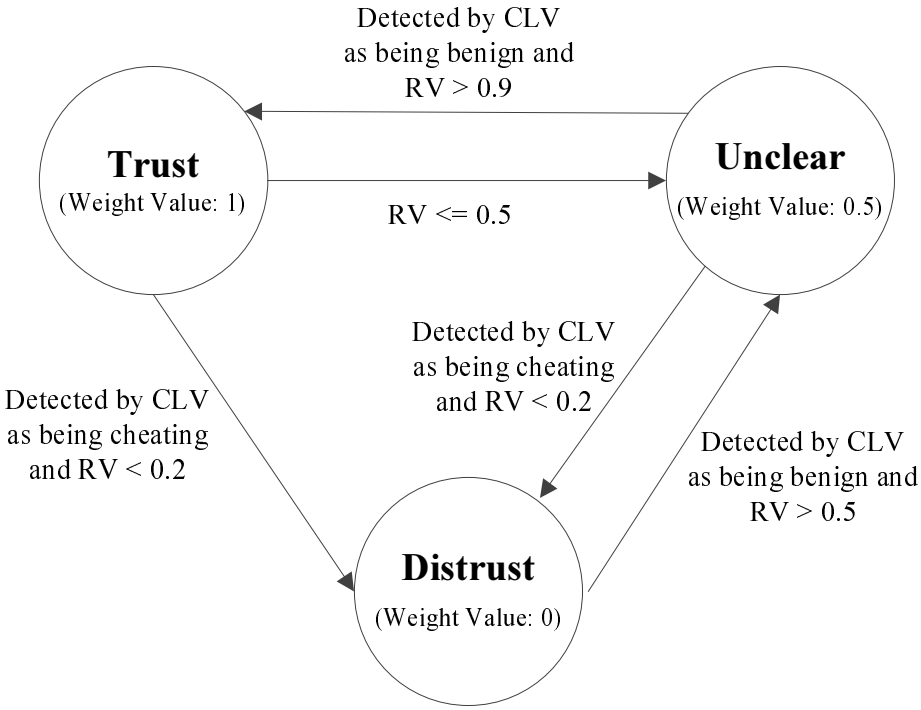

Figure 4.1: Reputation Management

applying CLV if that neighbor is in its Trust Category. Thus, we can reduce the overhead of the network. Certainly, vehicles in the Trust Category may start to cheat, hence the RV of the vehicles in the Trust Category will attenuate as time passes away. Therefore, after a vehicle stays in Trust category for a period of time (when RV<0.5), it will be moved into the Unclear Category. For example, if one vehicle stays in Trust category for 60 seconds, its RV will be reduced by 0.1. Certainly, vehicles do not consider those vehicles in the Distrust Category and the RV of such vehicles will increase as time goes by. When a RV increases above 0.5, it will be classified in the Unclear Category.

The Reputation Value (RV) is calculated in two ways: local calculation and globe calculation. For local calculation, vehicles will adjust their RV record every time when they apply CLV to other vehicles. If a target vehicle is detected as a malicious node, then its new RV is calculated by using the following formula:

$$RV_{new} = \frac{RV_{old}}{2} \tag{4.1}$$

If the target vehicle is determined to be benign, then its new RV is calculated by using the following formula:

$$RV_{new} = \begin{cases} 1, & if\ RV_{old} \times 1.5 > 1 \\ RV_{old} \times 1.5, & otherwise \end{cases} \tag{4.2}$$

Obviously, the RV slips fast if the vehicle is detected as falsifying its information. On the contrary, it's difficult to accumulate reputation.

As for the global calculation, we assume that a routing algorithm requires vehicles to broadcast their location information regularly. In order to calculate global reputation, every vehicle needs to attach their RV records to such broadcast messages. When a vehicle (V) receives this broadcast information (vehicles only collect the broadcast information within 2-3 hops), V will update its own RV record by using the following formula:

$$RV_i = \frac{\sum_{j \in S}(w_j \times RV_{ji})}{\sum_{j \in S} w_j} \tag{4.3}$$

where RV_i is the global reputation value for vehicle i recorded by V, and S is the set of vehicles whose RV records are included in the broadcast information (including vehicle V). w_j is the weight value of vehicle j maintained by vehicle V and RV_{ji} is the local RV of vehicle i rated by vehicle j. We set, respectively, weight value 1, 0.5 and 0 to Trust, Unclear and Distrust Categories (as Fig. 4.1 shows). For example, vehicle A received the broadcast RV records from vehicle B and B is in the Unclear Category of A's local records; therefore, the weight value of RV records from B will be considered as 0.5 by A.

4.2 Reputation History

Obviously, it will be better if the RMS maintains the reputation history for every vehicle. Besides, because the behaviors of vehicles may chance some time, we propose that the reputation history of RV is divided into many slots based on time. For example, if the

duration of time slot is 5 minutes, then the history table of one vehicle within the last one hour will have 12 reputation values which are irrelevant to each other. At the beginning of every time slot, the system will reset the vehicle's reputation value to 0.5. In other words, the reputation values recorded in the history table for each time slot are irrelevant to each other. Thus, the RMS can detect if one vehicle changes its behavior suddenly. Besides, if the network is not active enough to generate a reliable reputation value for judging a vehicle in the current time slot, the SPRT also can help to judge it by utilizing the history table based on the Sequential Probability Ratio Test (SPRT).

4.2.1 Overview of SPRT

Considering two hypotheses H_0 and H_1 which cannot be both true or false, the SPRT is used to determine which hypothesis is true based on a random sample. Such a random sample may not be enough to make a decision, which traditional hypothesis testing approaches do not consider. However, SPRT will continue to collect samples if the samples observed before are not sufficient to make an accurate decision [68].

In SPRT, for any positive integral value n, the probability that a sample $x_1, x_2, ..., x_m$ may be obtained is given by

$$P[x|H_1] = P[x_1|H_1] \times P[x_2|H_1] \times \cdots \times P[x_n|H_1] \tag{4.4}$$

when H_1 is true, and by

$$P[x|H_0] = P[x_1|H_0] \times P[x_2|H_0] \times \cdots \times P[x_n|H_0] \tag{4.5}$$

when H_0 is true [68].

The SPRT for testing H_0 against H_1 can then be described as follows: two positive constants A and B (B<A) are chosen. If $B < \frac{P[x|H_1]}{P[x|H_0]} < A$, the SPRT continues to take additional observations. If $\frac{P[x|H_1]}{P[x|H_0]} \geq A$, the experiment is terminated with the acceptance of H_1 (rejection of H_0). If $\frac{P[x|H_1]}{P[x|H_0]} \leq B$, we accept H_0 and terminate the test.

As suggested by [68], the values of A and B are bounded by $A \leq \frac{1-\beta}{\alpha}$ and $B \geq \frac{\beta}{1-\alpha}$, where α is the limit of the probability of accepting H_1 when H_0 is true in fact, and β is

the limit of the probability of accepting H_0 when H_1 is true in reality. The paper also proposed that the test can provide adequate level of accuracy by taking $A =\leq \frac{1-\beta}{\alpha}$ and $B =\geq \frac{\beta}{1-\alpha}$.

4.2.2 SPRT for Counting Reputation

As mentioned before, we can utilize the history of reputation values to decide whether one vehicle performs as a benign node or not in the current time slot. Let H_0 be the hypothesis that a given node is cooperative and H_1 be the hypothesis that a given node is malicious. Let $P[RV(j,i)_n|H_1]$ be the conditional probability of accepting H_1 and $P[RV(j,i)|H_0]$ be the conditional probability of accepting H_0, where $RV(j,i)_n$ denotes the RV of node i recorded by node j within the time slot n. Then we have:

$$P[RV(j,i)_n|H_1] = (1 - RV(j,i)_1) \times \cdots \times (1 - RV(j,i)_n) \tag{4.6}$$

$$P[RV(j,i)_n|H_0] = RV(j,i)_1 \times \cdots \times RV(j,i)_n \tag{4.7}$$

If $\frac{P[x|H_1]}{P[x|H_0]} \leq B$, we accept H_0 and terminate the test. We accept H_1 if $\frac{P[RV(j,i)_n|H_1]}{P[RV(j,i)_n|H_0]} \geq A$. And we accept H_0 if $\frac{P[RV(j,i)_n|H_1]}{P[RV(j,i)_n|H_0]} \leq B$. Then the process is terminated.

Clearly, such a process is similar to what has been described in the overview of SPRT. However, in our practical computation, we take the log-transformation of all parameters in equations (4.6) and (4.7). Then the $\log \frac{P[RV(j,i)_n|H_1]}{P[RV(j,i)_n|H_0]}$ can be written as:

$$\begin{aligned}
&\log \frac{P[RV(j,i)_n|H_1]}{P[RV(j,i)_n|H_0]} \\
&= \log \frac{1 - RV(j,i)_1}{RV(j,i)_1} + \cdots + \log \frac{1 - RV(j,i)_n}{RV(j,i)_n} \\
&= \sum_{m=1}^{n} \log \frac{1 - RV(j,i)_m}{RV(j,i)_m}
\end{aligned} \tag{4.8}$$

In conclusion, if

$$logB < \sum_{m=1}^{n} \log \frac{1 - RV(j,i)_m}{RV(j,i)_m} < \log A \tag{4.9}$$

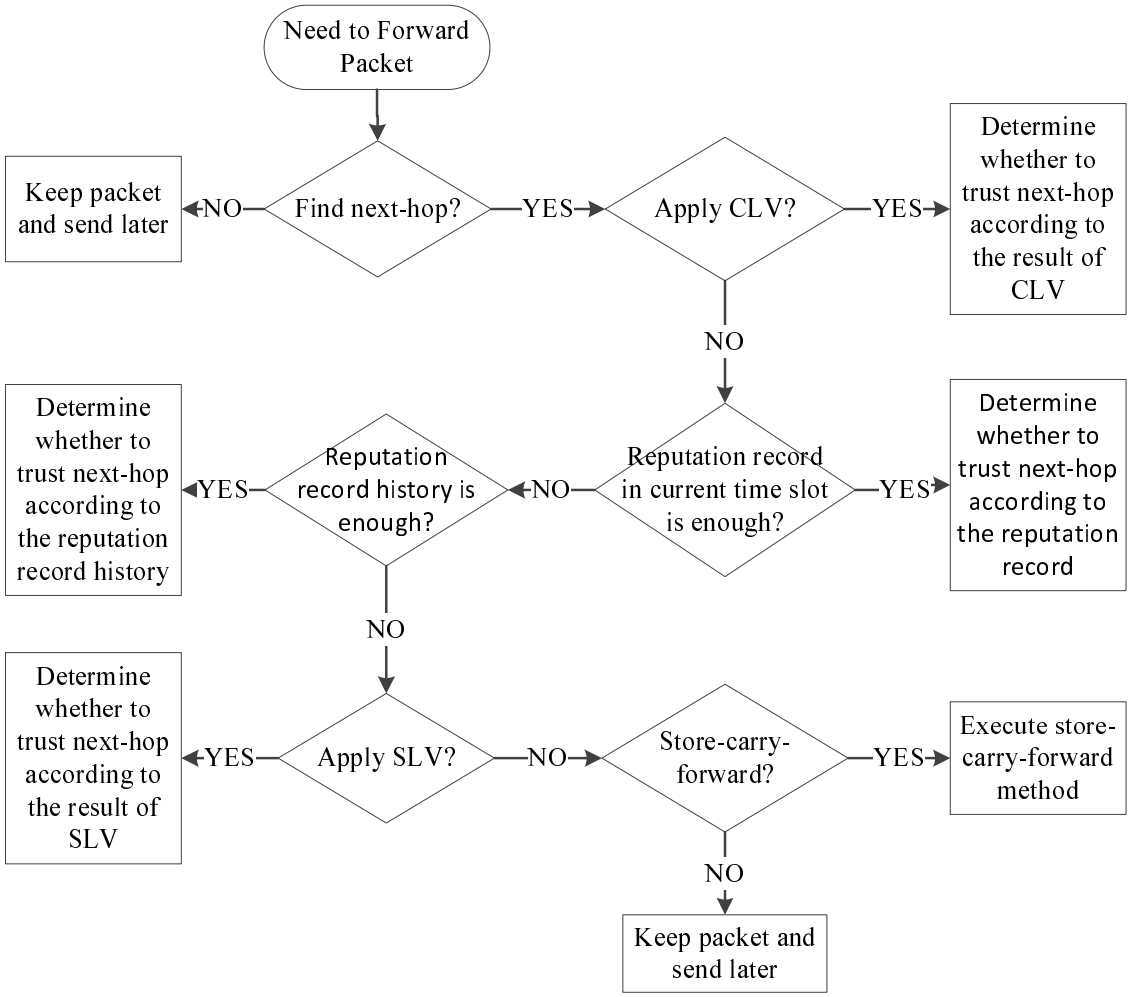

Figure 4.2: Handle Sparse Network

the test is continued to for gathering of additional observations. If

$$\sum_{m=1}^{n} \log \frac{1 - RV(j,i)_m}{RV(j,i)_m} \geq \log A \tag{4.10}$$

the test is terminated with the acceptance of H_1. If

$$\sum_{m=1}^{n} \log \frac{1 - RV(j,i)_m}{RV(j,i)_m} \leq \log B \tag{4.11}$$

the test is terminated with the acceptance of H_0.

4.3 Sparse Scenario

In real life, vehicles sometimes cannot find any other Cooperator to execute CLV, when vehicles need to send packets, because the density of vehicles in the road may not be great enough. Therefore, it is necessary to consider such sparse scenarios so that the proposed protocol can deal with it.

The proposed solution for the sparse networks be described as follows (see Fig. 4.2): when a vehicle needs to forward a packet, if it cannot find any other vehicle more superior than itself based on the routing algorithm, it has to keep the packet. It will try to send the packet a few seconds later. If the vehicle finds the next-hop but cannot find a Cooperator or Transmitter to execute CLV, it can determine whether to trust the next-hop according to its Reliability Category in the current time slot. If the relevant Reliability records are not enough to make a decision, it can rely on the reputation history. If the reputation history is not enough, it can apply the Secure Location Verification (SLV) [64], which requires a relatively low density of vehicles and, if it cannot even apply the SLV to complete forwarding the packets because of the sparse density it can keep the packet if it moves towards the destination. Otherwise, the vehicle can send the packet to one neighbor who is in the Trust Category and moving towards the destination (such store-carry-forward method has been used in [50]). If none of the conditions mentioned above can be applied, the vehicle has to keep the packet and send it later.

We can see an example in Fig. 4.3. Vehicle A cannot find an appropriate vehicle to forward the packet, so it chooses vehicle A which is moving towards the destination. After that, vehicle A carries the packet until it finds vehicle B.

4.4 Summary

This chapter shows a novel reputation management system. The first section explains how the RMS generates and maintains the reputation values for every vehicle. The vehicles can utilize the results of executing the CLV as the first-hand experience to

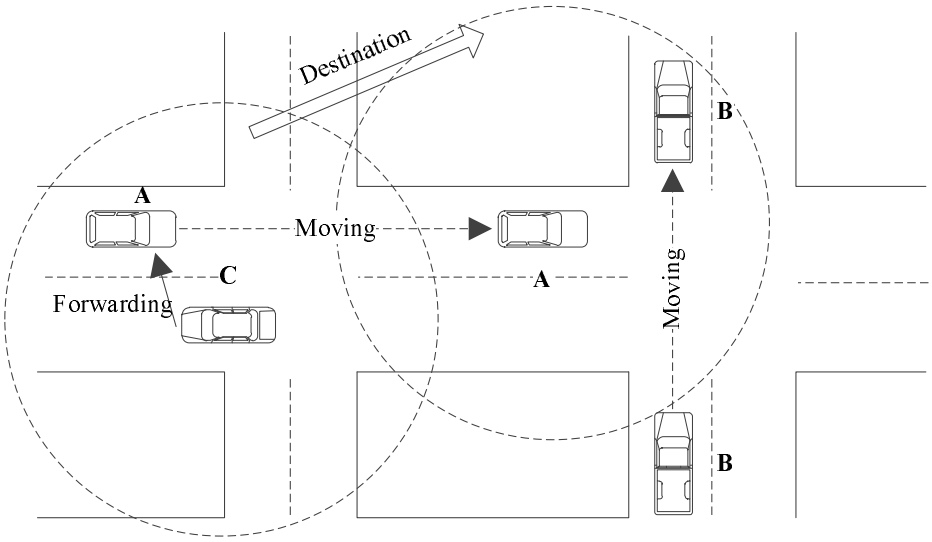

Figure 4.3: Carry and Forward

other vehicles. Besides, vehicles also can exchange the local reputation values of other vehicles with neighbors to learn others' records. The second section discusses the vehicles maintain the reputation history tables to keep the history of the reputation values. The history is divided into many slots by time. The RVs in every slot are totally irrelevant. The the system can use SPRT to combine the RVs to estimate the RV in the current time slot. In the next section, because the density of vehicles is low sometimes, a brief discussion about how to handle the sparse networks is given.

Chapter 5

Simulation

The core of the simulation is using NS-2 (Network Simulator 2). The simulation includes highway scenario and urban scenario and based on Greedy Forwarding Algorithm.

5.1 Simulation Parameters

Table 5.1 shows the simulation parameters in NS-2. For the mobility model of a highway, we used a 6 lane, two-direction road (3-lanes for each direction). The topology of the highway is 2000m long, and the lane width is 5m. The speed of vehicles is limited between 10 and 27.8 m/s. When the vehicle arrives at the end of the road, another end of the road will automatically generate one vehicle to ensure that there are always 100 vehicles running in the road. For the traffic model, we plot 5 vehicles sending packets to random destinations every 2 seconds and the transmission radio range is 250m. The propagation model is the two-ray ground reflection model.

The urban scenario is similar with a highway scenario except that the velocity of vehicles is limited to a range between 1 and 13.9 m/s and the topology is different. As shown in Fig. 5.1, for the topology of an urban scenario(800m×800m), we used an area that contains six roads and each road has two lanes (one lane for each direction). In both of these scenarios, malicious nodes will discard any packets they get and we added different number of malicious nodes to the scenarios to research various situations.

Table 5.1: Simulation and Experiments Parameters

Parameter	**Value Setting**
Radio Propagation	Two Ray Ground
Antenna Type	OmniAntenna
MAC Protocol	IEEE 802.11p
Routing Protocol	Greedy Forwarding Algorithm
Speed Limits (Highway)	10 - 27.8 m/s
Speed Limits (Urban City)	1 - 13.9 m/s

5.2 Evaluation Metrics

5.2.1 Packet Delivery Ratio

This parameter is the ratio of the number of successful delivery packets to the total number of packets that have been sent. Obviously, our target is to increase this ratio.

5.2.2 Packet Loss Ratio

Because malicious nodes will discard any packet they obtain in our simulation, the ratio of the number of packets that have been discarded to the total number of packets that have been sent is considered as Packet Loss Ratio (PLR). This information can complement the packet delivery ratio to evaluate the effectiveness of avoiding malicious nodes by using our algorithm.

5.2.3 Successful Detection Ratio

This parameter is the ratio of the number of malicious nodes being detected successfully to the total number of malicious nodes being verified (including success or failure to detect). The greater this number is, the higher the secure level is.

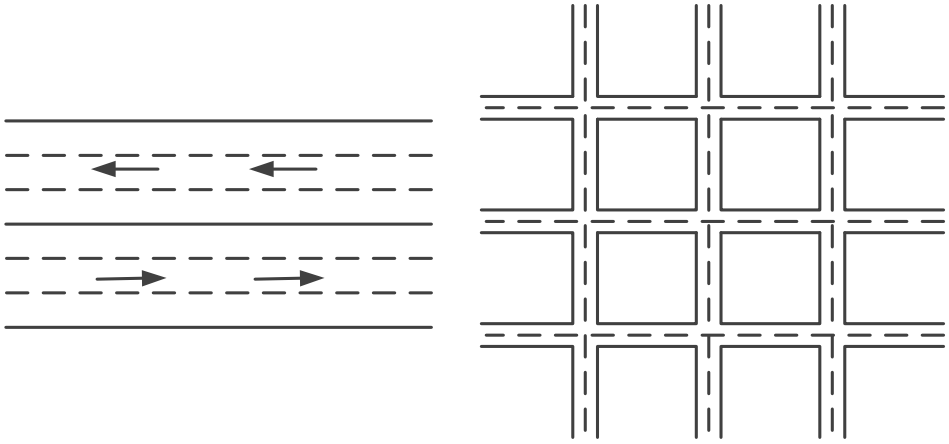

Figure 5.1: Highway and Urban Scenario

5.2.4 Average Verification Time

The verification time is the duration from the beginning of the verification to obtaining the verification results. The Average Verification Time (AVT) is the value of that the sum of time used to execute the location verification algorithm is divided by the number of times that the algorithm is executed. This parameter can reflect the network overhead.

5.2.5 Average Reputation Value

We use the Average Reputation Value (ARV) to evaluate the effectiveness of the proposed reputation management system. In the simulation, the ARV of benign vehicles and malicious vehicles are calculated respectively. This parameter is averaged over the reputation records collected from every vehicles in a specific time.

5.3 Evaluation of CLV

Table 5.2 shows the simulation parameters in the simulation to evaluate CLV. In order to prove the validity of CLV, I compared it to SLV in both highway scenario and urban scenario.

Table 5.2: Simulation Parameters for Evaluation of CLV

Parameter	**Value Setting**
Radio Range	250m
Beacon Freq	0.5 Hz
Number of Vehicles	100
Road (Highway)	2000m×30m
Area (Urban City)	800m×800m
Simulation Time	1000 - 1400s

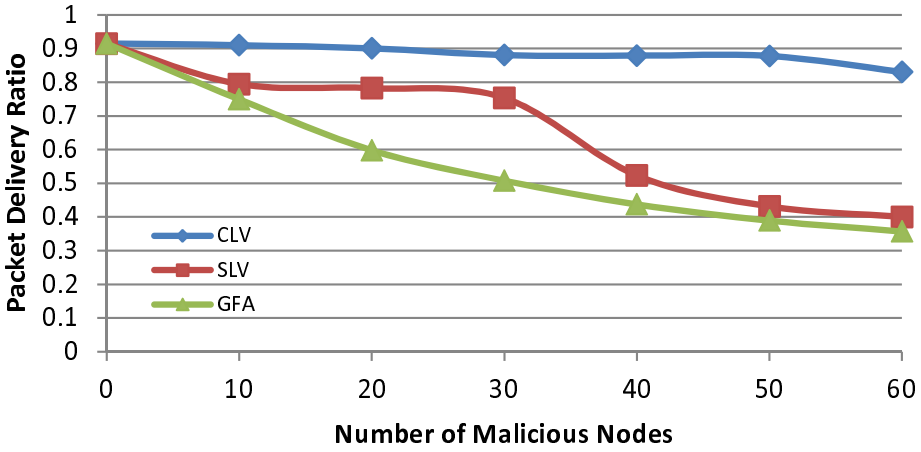

Figure 5.2: Package Delivery Ratio for Highway Scenario

As the results show, Our approach has a better performance than both SLV and GFA. In the simulation, an attacker will report its location with the enlargement of 15m.

5.3.1 Packet Delivery Ratio of CLV and SLV

As the Fig. 5.2 and Fig. 5.3 show, the proposed algorithm (CLV) can lead a higher Packet Delivery Ratio (PDR) than the SLV and GFA. Particularly in a highway scenario, when the number of malicious nodes exceeds 30, the proposed algorithm still maintains a stable PDR, when compared to the SLV whose curve presents a sudden drop. Note that the

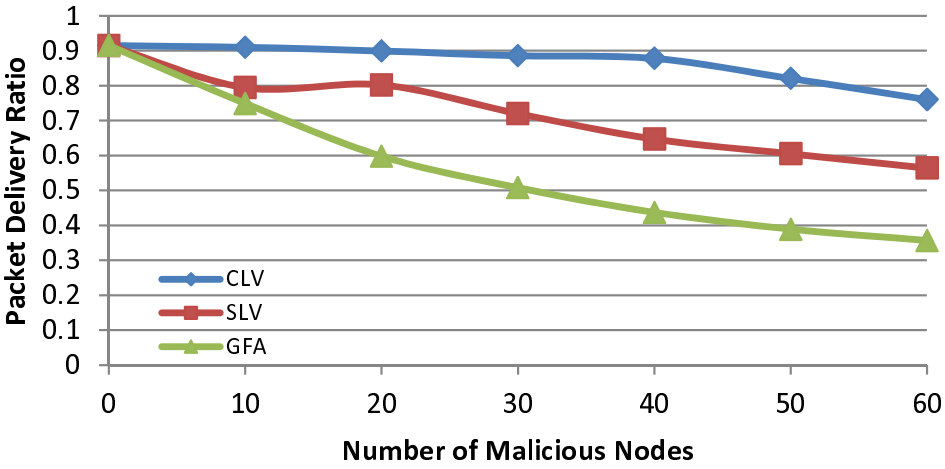

Figure 5.3: Package Delivery Ratio for Urban Scenario

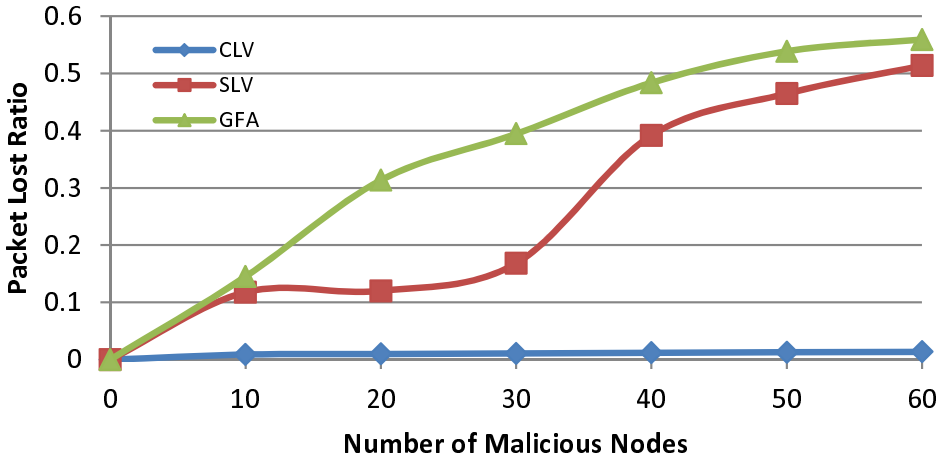

Figure 5.4: Packet Loss Ratio for Highway Scenario

PDR decreases as the number of malicious nodes increase. This is because it is that it's more possible for malicious nodes to receive packets when their numbers increase and they will inevitably discard the packet, therefore, failing to delivery the packets.

5.3.2 Packet Loss Ratio of CLV and SLV

The Fig. 5.4 and Fig. 5.5 show that the CLV has a lower Packet Loss Ratio than the SLV and the GFA. In the simulation, the malicious nodes will discard any packet they

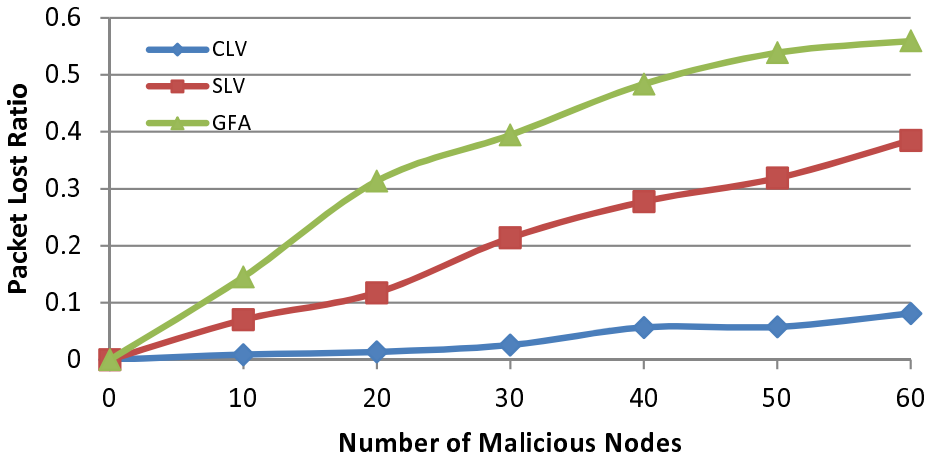

Figure 5.5: Packet Loss Ratio for Urban Scenario

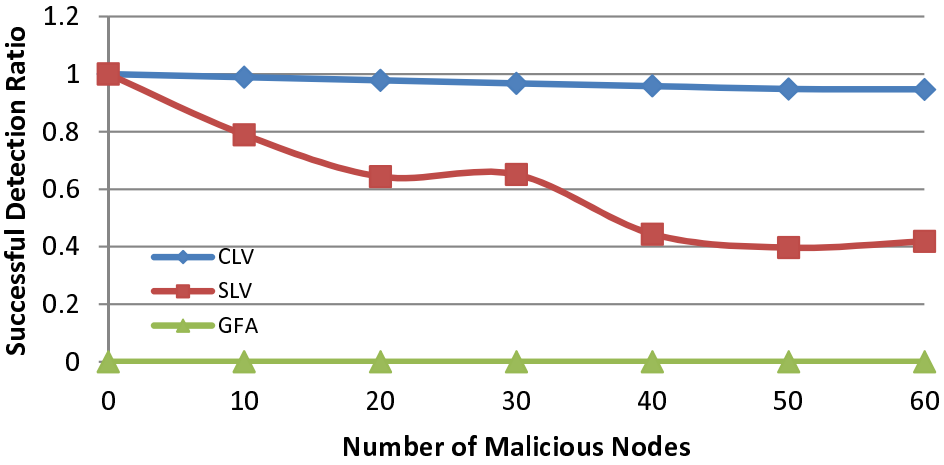

Figure 5.6: Successful Detection Ratio for Highway Scenario

receive. Hence, the Packet Loss Ratio can reflect this effectiveness to avoid malicious nodes. Certainly, the greater number of packets received by malicious nodes, the lower the PLR will be. Therefore our figures demonstrate that the CLV can more effectively prevent malicious nodes from obtaining packets than both the SLV and GFA.

5.3.3 Successful Detection Ratio of CLV and SLV

The Fig. 5.6 and Fig. 5.7 show that the CLV can achieve a higher Successful Detection Ratio (SDR). This directly reflects that the CLV has a larger probability of success for

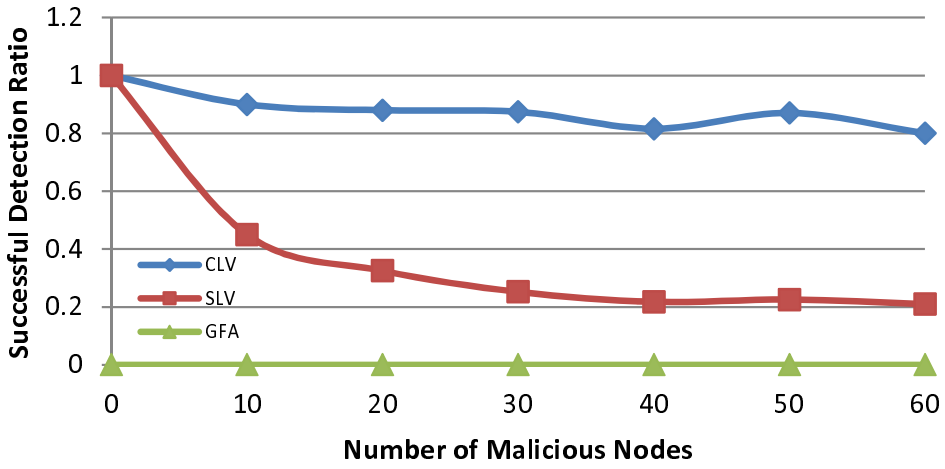

Figure 5.7: Successful Detection Ratio for Urban Scenario

the detection of malicious nodes than the SLV. In order to explain this parameter more clearly, here is an example: one vehicle needs to verify malicious nodes 100 times in total; however, it detects successfully just 60 times and fallaciously regards malicious nodes as benign nodes in the 40 other times. Obviously, the larger the SDR is, the more effective the algorithm will be. As for the GFA, its SDR constantly equals zero because it does not apply any location verification algorithm.

5.3.4 The Impact of Vehicles' Density to CLV and SLV

This simulation compares the CLV protocol to the SLV in the scenarios with different densities of vehicles. The density of vehicles is estimated by the average distance between vehicles. In Fig. 5.8, as the density of vehicles (15 percent of the vehicles are malicious) increases, the PDR grows and finally maintains a high level. This figure also demonstrates that our protocol performs better than SLV in the different density scenario. In the urban scenario (Fig. 5.9), we obtained similar results to the highway scenario. However, the urban scenario required a lower density to achieve a higher PDR than the highway scenario. As far as this thesis author is concerned, this is because the roads are dense in the urban scenario so that vehicles can depend on vehicles locating in other roads. This

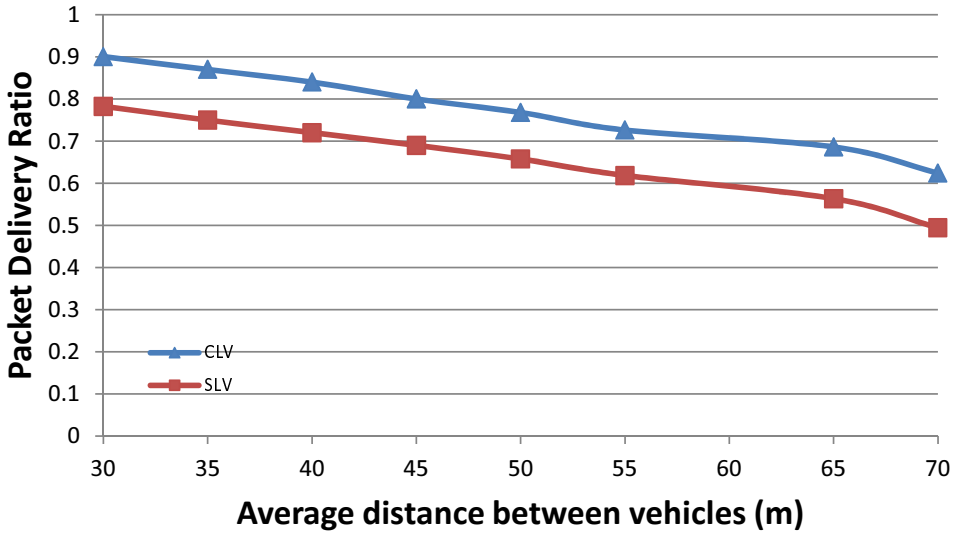

Figure 5.8: Impact of Density in Highway

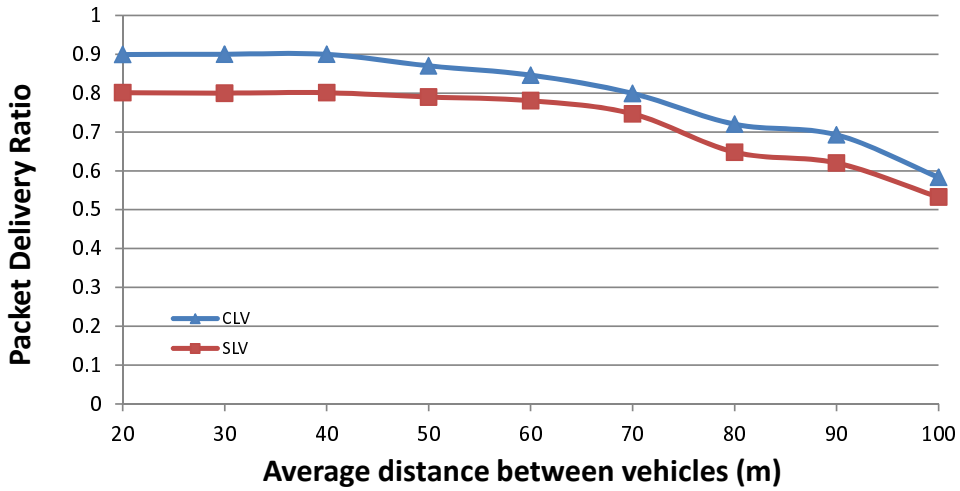

Figure 5.9: Impact of Density in Urban

subject can be a topic of the future research works.

5.3.5 Average Verification Time of CLV and SLV

Furthermore, I utilize the AVT to compare our algorithm (CLV) to SLV. In this simulation, the verification time is calculated for CLV without using the Reputation Management System; hence this can show the comparison of the natures of CLV and SLV. As

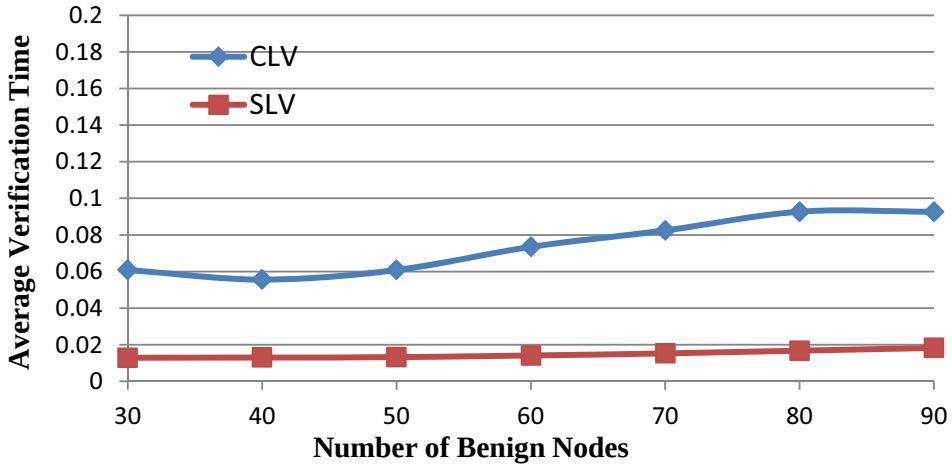

Figure 5.10: Average Verification Time for Highway Scenario

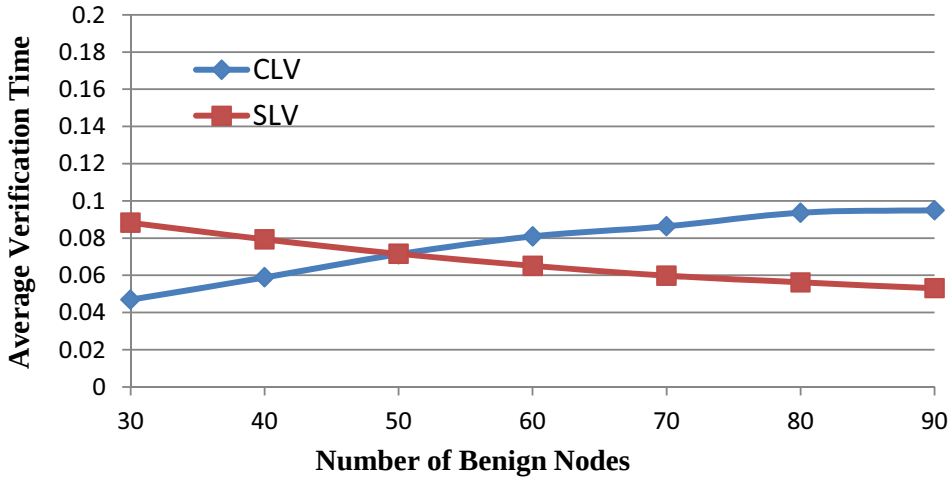

Figure 5.11: Average Verification Time for Urban Scenario

the Fig. 5.10 shows, the AVT of CLV is approximately tripe as long as the AVT of SLV with a small number of benign nodes; and it is about four times the duration of the AVT of SLV with the larger number of nodes. It has been expected before the simulation that CLV has a larger AVT than SLV and its AVT increases as the number of nodes grows.

The AVT in urban scenario is interesting because the AVT of CLV increases while the AVT of SLV decreases as the number of benign nodes grows (see Fig. 5.11). The reason of such situation occurs may be as follows: the selection of the Cooperator in CLV

Table 5.3: Simulation Parameters for Evaluation of Reputation Management

Parameter	**Value Setting**
Radio Range	250m
Beacon Freq	0.5 Hz
Number of Vehicles (Highway)	50
Number of Vehicles (Urban)	25
Road (Highway)	1000m×30m
Area (Urban City)	500m×500m
Simulation Time	6000s

is stricter than the rules in SLV; therefore, the greater the number of nodes is, the easier to find an appropriate Cooperator is in SLV. However, the CLV needs to compare the information of all the candidates to select the most suitable one; hence the verification time will increase as the number of nodes rises. Optimizing the verification time of CLV could be researched in the future work.

5.4 Evaluation of Reputation Management

In this section, most of the simulations for the RMS are evaluated by several different metrics and the simulations utilize the parameters shown in the Table 5.3.

5.4.1 Reputation Management System without using SPRT

In the simulation, the beneficial ramifications of the reputation system in the whole scheme have been studied. To evaluate such benefits, we used the ratio of number of times of verification with the reputation management system to the number of times of verification without the reputation management system in the same scenarios. The highway scenario is constructed as $2000m \times 30m$, as mentioned before, and there are 100 vehicles moving in the roads which contain 30 malicious vehicles. The result shows

that the ratio is approximately 40 percent; this means that the reputation management system required 60 percent less verifications to achieve a similar result when compared to the scheme without a reputation management system. Similarly, in the urban scenario ($800m \times 800m$) which also contains 70 benign vehicles and 30 malicious vehicles, the simulation shows that the ratio is about 50 percent which means the RMS saves 50 percent of time used in the verification. These results prove the benefit and necessity of having the reputation management system.

In this simulation to evaluate ARP, the topology of $2000m \times 30m$ is used as the highway scenario and there are 100 vehicles moving in it including 30 malicious nodes; the topology of $800m \times 800m$ is used as the urban scenario and there are 50 vehicles moving in it including 20 malicious nodes; and the simulation time is 1000s. In Fig. 5.12, the Average Reputation Value (ARP) of benign nodes increases dramatically at the beginning of the simulation and maintains this level. On the contrary, the ARP of malicious nodes decreases at the end of the simulation as expected. As mentioned in the section of Reputation Management, vehicles in the Trust and Distrust Categories will not be verified for the purpose of maintaining a lower network overhead. Therefore, the ARP of benign nodes does not grow to 1 and the ARP of malicious nodes does not fall to 0. In Fig. 5.13, we obtained a similar simulation result in the urban scenario; this further demonstrated the beneficial nature of our reputation management system and we will continue to improve its ability to adapt in our future work.

In the previous simulation, the malicious nodes inevitably report the wrong position information. However, in the real life, the malicious nodes may not cheat all the time. Hence it is necessary to explore how the reputation works when the malicious nodes cheat by chance. In other word, it is necessary to find the threshold of the cheating probability to attract the notice of the RMS.

In the simulation, I set different percentages ranging from 20 percent to 100 percent as the probability of cheating. Obviously, such percentages just affect the reputation value of the malicious nodes; therefore, the reputation values will maintain approximately

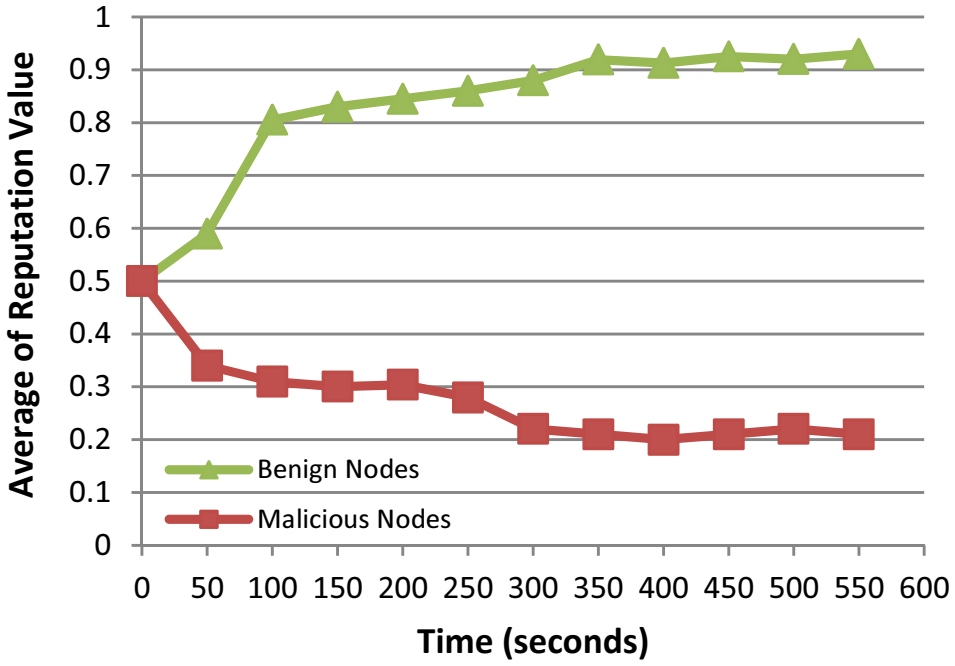

Figure 5.12: Result of Reputation System for Highway

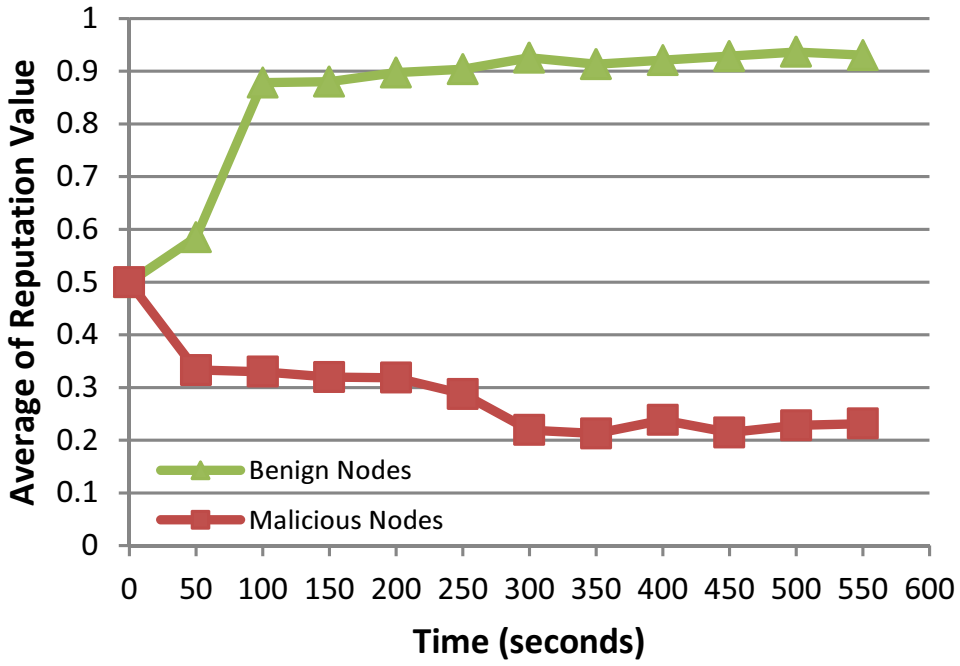

Figure 5.13: Result of Reputation System for Urban

stable. As simulation result of the highway shown in the Fig. 5.14, the top line shows the RV of the benign nodes and the other lines show the RVs of the malicious nodes with varying probabilities to cheat. The Average Reputation Value (APV) of the malicious nodes increases as probability of cheating decreases. By analyzing the figure, 60 percent seems to be a threshold for the RMS to distinguish the malicious node; because the RV will closer to the RV of the benign nodes when the percentage is lower than 60 percent. Obviously, it is convenient if there is a function to calculate such threshold. For example, we input the value of the required probability that the nodes cheat; the nodes whose cheating probabilities are higher than such predefined probability will be considered as malicious nodes. The function can output a reputation value for us to be used as the threshold to distinguish the malicious nodes from benign nodes. Such function could be researched in my future work.

In the urban scenario, the similar results are obtained as Fig. 5.15 shows.

5.4.2 Reputation Management System with using SPRT

In order to prove the effectiveness of using reputation history, I design anther experiment to find out whether the reputation history can help the verification when the reputation value in the current time slot is not enough to judge a node. The value of parameters can be seen in Table 5.3. For example, in one simulation for the highway scenario, there are 50 vehicles including 10 malicious vehicles. I set both the α and β as 0.15 and the duration of one time slot is 300 seconds. The result shows that the reputation value in the current time slot fails to help the verification by about 8400 times during the simulation (6000s); and the reputation history succeed to judge the nodes by approximately 5600 times in the 8400 times. Therefore, the reputation history can improve the RMS without applying the reputation history by 66 percent.

As the Fig. 5.16 shows, the percentage improved by the reputation history for the RMS slightly grows as the value of α and β increases, because the less accuracy the system requires, the less the data needed by the SPRT to obtain the result will be. In

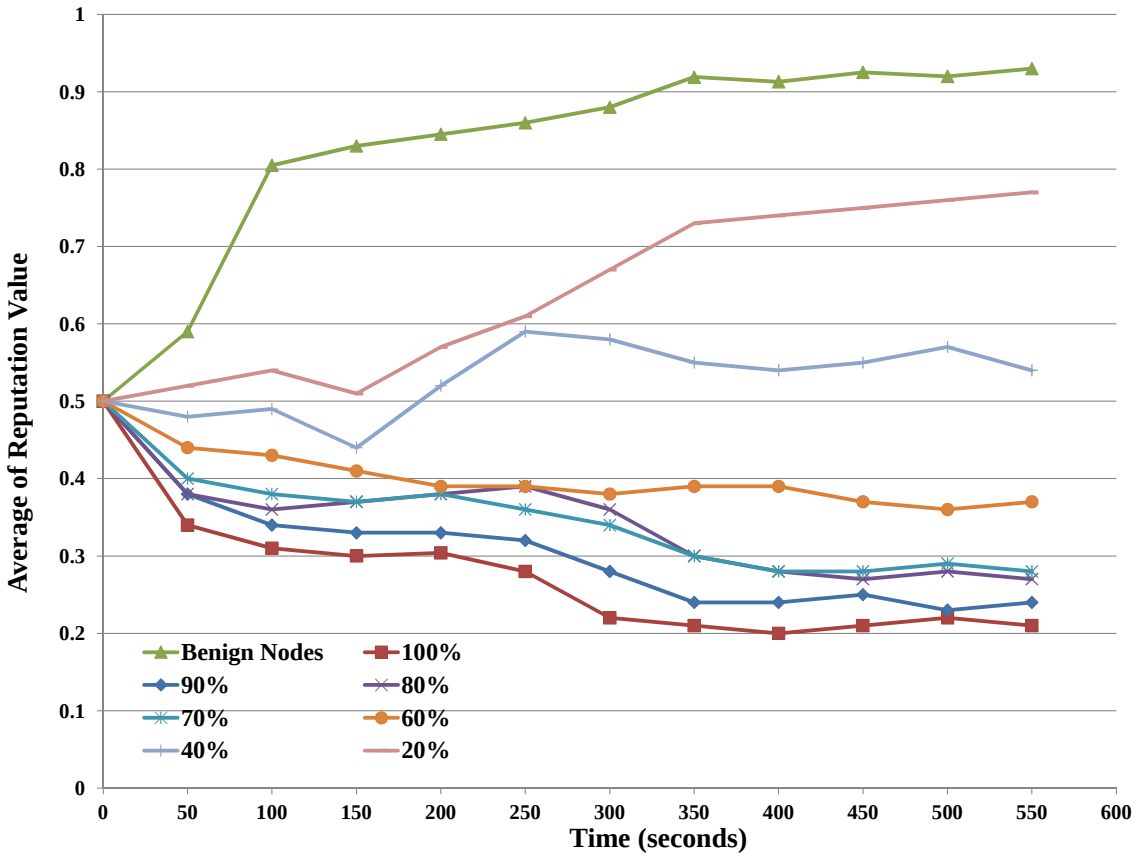

Figure 5.14: Nodes Cheating by Chance for Highway

the future, we can find a most appropriate values for the α and β.

5.5 Summary

This chapter show the experiments to prove the validness of the Cooperative Location Verification and the Reputation Management System. The first section shows the values of the general simulation parameters. In the section, several evaluation metrics are introduced which are used to evaluate the CLV and the RMS. After that, it shows the results of the evaluation of the CLV and the analyses of every result are provided. The results show that the CLV performs better than the SLV in the aspects of security. However, I still need to improve the verification time of the CLV in the future. In the

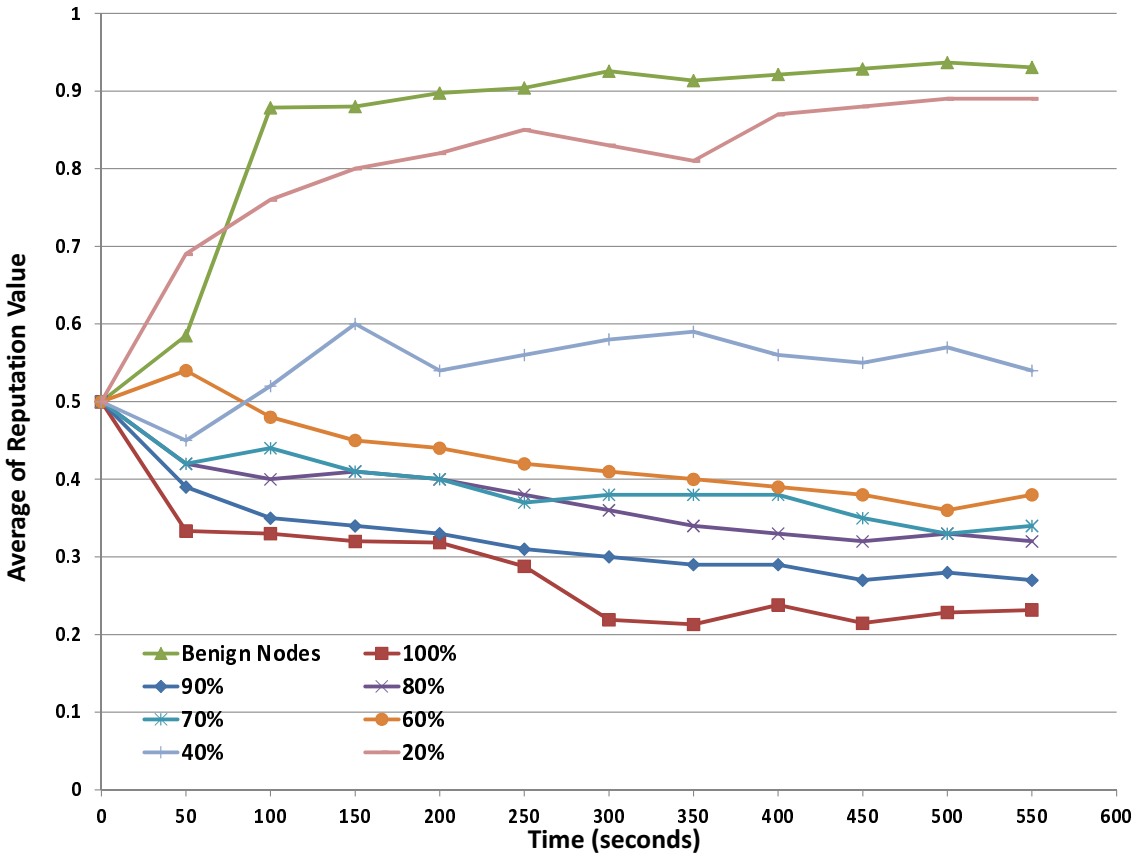

Figure 5.15: Nodes Cheating by Chance for Urban

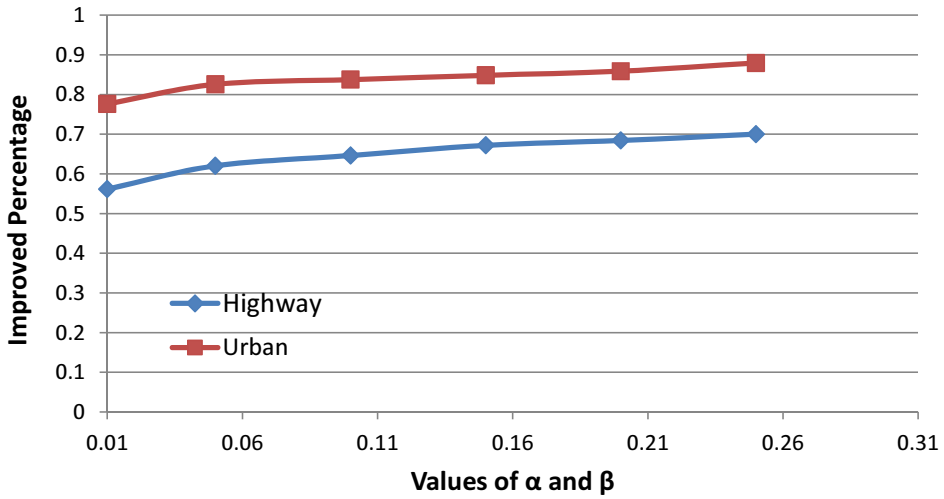

Figure 5.16: Benefits of Reputation History with Varying α, β

next section, the simulation results of the RMS are shown and discussions are given. Although I did not compare my RMS to other trust models, I provide the proof that my RMS can improve the effectiveness of the CLV by decreasing the network overhead. I also prove that the using of the reputation history enhances the RMS.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, a location verification algorithm, namely Cooperative Location Verification Protocol, is designed for VANETs to prevent the position-spoofing attack. In addition, a reputation management system to enhance the CLV is proposed; the RMS includes the generation and the history of reputation values. Finally, the results of the simulation show that the security level of the CLV is higher than that of the SLV, and the effectiveness of the RMS.

The CLV is mainly based on challenge-response procedure, where a receiver cannot viciously reduce the distance between itself and the sender. In order to prevent a vehicle from enlarging the distance by intentionally delaying the response, a series of rules to select the Cooperator are created to help the verification of a Prover's claimed location. Finally, the Verifier determines whether to trust a Prover's claimed location based on both the Verifier and the Cooperator's verification results. A series of rules for selecting the best Cooperator and the Transmitter has been produced. Then the effectiveness of the CLV is proved by the mathematical method.

In addition, a reputation management system is designed to maintain the results of verifications so that it can effectively guarantee the security and decline the network's overhead. The reputation managements system will generate a reputation value for every

vehicle according to the verification results of the CLV. The reputation history is divided into many slots based on time; those time slots are irrelevant. And if a vehicle cannot judge another node based on the reputation value in the current slot, the reputation history could help based on the SPRT.

In the simulation, the designed algorithm (CLV) is compared to the Secure Location Verification (SLV). The simulation results show that our algorithm has a higher security level. As the number of malicious nodes increases, the benefit provided by our algorithm becomes more and more distinct. The effectiveness of the reputation management system is also demonstrated in the simulation.

6.2 Future Work

6.2.1 Adaptive Calculation of Reputation Value

As mentioned in the section of the simulation, the malicious nodes are usually tricky and will not always report a wrong location; therefore the RMS needs a function to calculate the threshold of the reputation value to distinguish the malicious nodes according to the tolerance of the error in the networks. For example, one network is very strict and allows a few mistakes of claiming a unreal position. Then the RMS should adapt to it by taking a few changes.

6.2.2 Optimize the Verification Time of CLV

In the simulation, it is realized that the verification time of the CLV is higher than the SLV. In order to reduce the risk of leading to much network overhead, the CLV still needs to be improved in the future. For instance, the rules of selection the Cooperator could be improved to be more efficient.

6.2.3 Integrate Other Verifications with CLV

In the future, some simple but effective and efficient verification methods can be integrated into the CLV. For example, before applying the CLV to verify the location information claimed by a node, the system can first check some simple conditions if there was another report of location information from the same node recently. Then it can check whether the current position exceeds the maximum distance it can travel from last position based on the time between the two reports and the maximum speed limit in the surrounding area.

6.2.4 Best Performance of the SPRT

As mentioned before, in the process of the SPRT, α is the limit of the probability of accepting hypothesis H_1 when H_0 is true in fact, and β is the limit of the probability of accepting H_0 when H_1 is true in reality. It is obvious that the less the values of α and β are, the more accurate the testing result will be. However, if the values are too small, the SPRT will need numerous samples to make a decision. Therefore the values of α and β should have an appropriate point in order to reduce the quantity of samples needed without affecting the reliability of the results.

Bibliography

[1] Cvis. http://www.cvisproject.org, Feburary 2012.

[2] Global positioning system (gps). http://en.wikipedia.org/wiki/Global_Positioning_System, Feburary 2012.

[3] Mobile ad-hoc network. http://www.fermentas.com/techinfo/nucleicacids/maplambda.htm, January 2012.

[4] Network simulator 2. http://nsnam.isi.edu/nsnam/index.php/Main_Page, Feburary 2012.

[5] Nserc diva. http://www.nsercdiva.com/index.php, Feburary 2012.

[6] Radio frequency. http://en.wikipedia.org/wiki/Radio_frequency, Feburary 2012.

[7] Sevecom. http://www.sevecom.org, Feburary 2012.

[8] O. Abumansoor and A. Boukerche. A cooperative multi-hop location verification for non line of sight (nlos) in vanet. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, pages 773–778. IEEE, 2011.

[9] J. Blum, A. Eskandarian, and L. Hoffman. Mobility management in ivc networks. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 150–155. IEEE, 2003.

[10] J.J. Blum, A. Eskandarian, and L.J. Hoffman. Challenges of intervehicle ad hoc networks. *Intelligent Transportation Systems, IEEE Transactions on*, 5(4):347–351, 2004.

[11] A. Boukerche. *Handbook of algorithms for wireless networking and mobile computing*, volume 8. CRC Press, 2006.

[12] A. Boukerche. *Algorithms and protocols for wireless and mobile ad hoc networks*, volume 77. Wiley-IEEE Press, 2009.

[13] A. Boukerche. *Algorithms and protocols for wireless sensor networks*, volume 62. Wiley-IEEE press, 2009.

[14] L. Briesemeister, L. Schafers, and G. Hommel. Disseminating messages among highly mobile hosts based on inter-vehicle communication. In *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, pages 522–527. IEEE, 2000.

[15] P. Cencioni and R. Di Pietro. A mechanism to enforce privacy in vehicle-to-infrastructure communication. *Computer Communications*, 31(12):2790–2802, 2008.

[16] C. Chen, X. Wang, W. Han, and B. Zang. A robust detection of the sybil attack in urban vanets. In *Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference on*, pages 270–276. IEEE, 2009.

[17] Wai Chen, Ratul K. Guha, Taek Jin Kwon, John Lee, and Irene Y. Hsu. A survey and challenges in routing and data dissemination in vehicular ad-hoc networks. In *IEEE International Conference on Vehicular Electronics and Safety*, pages 328–333, Columbus, OH, USA, September 2008.

[18] T.W. Chim, SM Yiu, L.C.K. Hui, and V.O.K. Li. Security and privacy issues for inter-vehicle communications in vanets. In *Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2009. SECON Workshops' 09. 6th Annual IEEE Communications Society Conference on*, pages 1–3. IEEE, 2009.

[19] Q. Ding, X. Li, M. Jiang, and X.H. Zhou. Reputation-based trust model in vehicular ad hoc networks. In *Wireless Communications and Signal Processing (WCSP), 2010 International Conference on*, pages 1–6. IEEE, 2010.

[20] Q. Ding, X. Li, M. Jiang, and X.H. Zhou. Reputation management in vehicular ad hoc networks. In *Multimedia Technology (ICMT), 2010 International Conference on*, pages 1–5. IEEE, 2010.

[21] F. Dotzer, L. Fischer, and P. Magiera. Vars: A vehicle ad-hoc network reputation system. In *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a*, pages 454–456. IEEE, 2005.

[22] M. Durresi, A. Durresi, and L. Barolli. Emergency broadcast protocol for inter-vehicle communications. In *Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on*, volume 2, pages 402–406. IEEE, 2005.

[23] H. Füßler, M. Mauve, H. Hartenstein, M. Käsemann, and D. Vollmer. Mobicom poster: location-based routing for vehicular ad-hoc networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(1):47–49, 2003.

[24] M. Gerlach. Trust for vehicular applications. In *Autonomous Decentralized Systems, 2007. ISADS'07. Eighth International Symposium on*, pages 295–304. IEEE, 2007.

[25] P. Golle, D. Greene, and J. Staddon. Detecting and correcting malicious data in vanets. In *Proceedings of the 1st ACM international workshop on Vehicular Ad Hoc Networks*, pages 29–37. ACM, 2004.

[26] Z.J. Haas, M.R. Pearlman, and P. Samar. The zone routing protocol (zrp) for ad hoc networks. 2002.

[27] H. Hartenstein and K.P. Laberteaux. A tutorial survey on vehicular ad hoc networks. *Communications Magazine, IEEE*, 46(6):164–171, 2008.

[28] J.P. Hubaux, S. Capkun, and J. Luo. The security and privacy of smart vehicles. *Security & Privacy, IEEE*, 2(3):49–55, 2004.

[29] JT Isaac, S. Zeadally, and JS Camara. Security attacks and solutions for vehicular ad hoc networks. *Communications, IET*, 4(7):894–903, 2010.

[30] D.B. Johnson, D.A. Maltz, J. Broch, et al. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad hoc networking*, 5:139–172, 2001.

[31] B. Karp and H.T. Kung. Gpsr: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM, 2000.

[32] G. Korkmaz, E. Ekici, F. Özgüner, and Ü. Özgüner. Urban multi-hop broadcast protocol for inter-vehicle communication systems. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 76–85. ACM, 2004.

[33] C. Langley, R. Lucas, and H. Fu. Key management in vehicular ad-hoc networks. In *Electro/Information Technology, 2008. EIT 2008. IEEE International Conference on*, pages 223–226. IEEE, 2008.

[34] T. Leinmüller, C. Maihöfer, E. Schoch, and F. Kargl. Improved security in geographic ad hoc routing through autonomous position verification. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 57–66. ACM, 2006.

[35] T. Leinmuller, E. Schoch, and F. Kargl. Position verification approaches for vehicular ad hoc networks. *Wireless Communications, IEEE*, 13(5):16–21, 2006.

[36] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer. Influence of falsified position data on geographic ad-hoc routing. *Security and Privacy in Ad-hoc and Sensor Networks*, pages 102–112, 2005.

[37] F. Li and Y. Wang. Routing in vehicular ad hoc networks: A survey. *Vehicular Technology Magazine, IEEE*, 2(2):12–22, 2007.

[38] X. Lin, R. Lu, C. Zhang, H. Zhu, P.H. Ho, and X. Shen. Security in vehicular ad hoc networks. *Communications Magazine, IEEE*, 46(4):88–95, 2008.

[39] B. Liu, Y. Zhong, and S. Zhang. Probabilistic isolation of malicious vehicles in pseudonym changing vanets. In *Computer and Information Technology, 2007. CIT 2007. 7th IEEE International Conference on*, pages 967–972. IEEE, 2007.

[40] G. Liu, B.S. Lee, B.C. Seet, C.H. Foh, K.J. Wong, and K.K. Lee. A routing strategy for metropolis vehicular communications. *Information Networking. Networking Technologies for Broadband and Mobile Networks*, pages 134–143, 2004.

[41] G. Liu, B.S. Lee, B.C. Seet, C.H. Foh, K.J. Wong, and K.K. Lee. A routing strategy for metropolis vehicular communications. *Information Networking. Networking Technologies for Broadband and Mobile Networks*, pages 134–143, 2004.

[42] Y. Liu, J. Bi, and J. Yang. Research on vehicular ad hoc networks. In *Control and Decision Conference, 2009. CCDC'09. Chinese*, pages 4430–4435. IEEE, 2009.

[43] N.W. Lo and H.C. Tsai. Illusion attack on vanet applications-a message plausibility problem. In *Globecom Workshops, 2007 IEEE*, pages 1–8. IEEE, 2007.

[44] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve. A routing strategy for vehicular ad hoc networks in city environments. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 156–161. Ieee, 2003.

[45] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein. Geographic routing in city scenarios. *ACM SIGMOBILE Mobile Computing and Communications Review*, 9(1):69–72, 2005.

[46] C. Maihofer and R. Eberhardt. Geocast in vehicular environments: Caching and transmission range control for improved efficiency. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 951–956. IEEE, 2004.

[47] C. Maihöfer, T. Leinmüller, and E. Schoch. Abiding geocast: time–stable geocast for ad hoc networks. In *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pages 20–29. ACM, 2005.

[48] S. Marti, T.J. Giuli, K. Lai, M. Baker, et al. Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 255–265, 2000.

[49] U.F. Minhas, J. Zhang, T. Tran, and R. Cohen. Towards expanded trust management for agents in vehicular ad-hoc networks. *International Journal of Computational Intelligence Theory and Practice (IJCITP)*, 5(1), 2010.

[50] H. Oka and H. Higaki. Multihop data message transmission with inter-vehicle communication and store-carry-forward in sparse vehicle ad-hoc networks (vanet). In *New Technologies, Mobility and Security, 2008. NTMS'08.*, pages 1–5. IEEE, 2008.

[51] B. Parno and A. Perrig. Challenges in securing vehicular networks. In *Workshop on Hot Topics in Networks (HotNets-IV)*, pages 1–6, 2005.

[52] A. Patwardhan, A. Joshi, T. Finin, and Y. Yesha. A data intensive reputation management scheme for vehicular ad hoc networks. In *Mobile and Ubiquitous Systems: Networking & Services, 2006 Third Annual International Conference on*, pages 1–8. IEEE, 2006.

[53] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on*, pages 90–100. IEEE, 1999.

[54] K. Plossl, T. Nowey, and C. Mletzko. Towards a security architecture for vehicular ad hoc networks. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, pages 8–pp. IEEE, 2006.

[55] M. Raya and J.P. Hubaux. Securing vehicular ad hoc networks. *Journal of Computer Security*, 15(1):39–68, 2007.

[56] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.P. Hubaux. Eviction of misbehaving and faulty nodes in vehicular networks. *Selected Areas in Communications, IEEE Journal on*, 25(8):1557–1568, 2007.

[57] M. Raya, P. Papadimitratos, V.D. Gligor, and J.P. Hubaux. On data-centric trust establishment in ephemeral ad hoc networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1238–1246. IEEE, 2008.

[58] M. Raya, P. Papadimitratos, and J.P. Hubaux. Securing vehicular communications. *Wireless Communications, IEEE*, 13(5):8–15, 2006.

[59] M.T. Refaei, L.A. DaSilva, M. Eltoweissy, and T. Nadeem. Adaptation of reputation management systems to dynamic network conditions in ad hoc networks. *Computers, IEEE Transactions on*, 59(5):707–719, 2010.

[60] Z. Ren, W. Li, and Q. Yang. Location verification for vanets routing. In *Wireless and Mobile Computing, Networking and Communications, 2009. WIMOB 2009. IEEE International Conference on*, pages 141–146. IEEE, 2009.

[61] Ghassan Samara, Wafaa A.H. Al-Salihy, and R. Sures. Security analysis of vehicular ad hoc networks. In *2010 Second International Conference on Network Applications, Protocols and Services*, pages 1–6, National Advanced IPv6 Center, Universiti Sains Malaysia Penang, Malaysia, 2010.

[62] RA Santos, A. Edwards, RM Edwards, and NL Seed. Performance evaluation of routing protocols in vehicular ad-hoc networks. *International Journal of Ad Hoc and Ubiquitous Computing*, 1(1):80–91, 2005.

[63] N. Sastry, U. Shankar, and D. Wagner. Secure verification of location claims. In *Proceedings of the 2nd ACM workshop on Wireless security*, pages 1–10. ACM, 2003.

[64] J.H. Song, V.W.S. Wong, and V.C.M. Leung. Secure location verification for vehicular ad-hoc networks. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1–5. IEEE, 2008.

[65] M.T. Sun, W.C. Feng, T.H. Lai, K. Yamada, H. Okada, and K. Fujimura. Gps-based message broadcasting for inter-vehicle communication. In *Parallel Processing, 2000. Proceedings. 2000 International Conference on*, pages 279–286. IEEE, 2000.

[66] L. Tian, Y. Zhou, and L. Tang. Improving gps positioning precision by using optical encoders. In *Intelligent Transportation Systems, 2000. Proceedings. 2000 IEEE*, pages 293–298. IEEE, 2000.

[67] Y. Toor, P. Muhlethaler, and A. Laouiti. Vehicle ad hoc networks: Applications and related technical issues. *Communications Surveys & Tutorials, IEEE*, 10(3):74–88, 2008.

[68] Abraham Wald. Sequential analysis. chapter 3. In *Dover Phoenix Editions*, The work originally was published by John Wiley and Sons Inc. in 1947, 2004.

[69] Y. Xi, K. Sha, W. Shi, L. Schwiebert, and T. Zhang. Enforcing privacy using symmetric random key-set in vehicular networks. In *Autonomous Decentralized Systems, 2007. ISADS'07. Eighth International Symposium on*, pages 344–351. IEEE, 2007.

[70] G. Yan, S. Olariu, and M.C. Weigle. Providing vanet security through active position detection. *Computer Communications*, 31(12):2883–2897, 2008.

[71] J. Zhang. A survey on trust management for vanets. In *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*, pages 105–112. IEEE, 2011.

[72] J. Zhang, C. Chen, and R. Cohen. A scalable and effective trust-based framework for vehicular ad-hoc networks. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 1(4):3–15, 2010.