PROVIDING LOCATION SECURITY IN VEHICULAR AD HOC NETWORKS

by

Gongjun Yan
B.S. June 1999, Sichuan Institute of Technology
M.S. June 2004, University of Electronic Science and Technology of China

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirement for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
May 2010

Approved by:

_____
Stephan Olariu (Director)

_____
Michele C. Weigle (Director)

_____
Kurt J. Maly (Member)

_____
Ravi Mukkamala (Member)

_____
Jinhua Guo (Member)

ABSTRACT

PROVIDING LOCATION SECURITY IN VEHICULAR AD HOC NETWORKS

Gongjun Yan
Old Dominion University, 2010
Co-Directors: Dr. Stephan Olariu
Dr. Michele C. Weigle

Location is fundamental information in Vehicular Ad-hoc Networks (VANETs). Almost all VANET applications rely on location information. Therefore it is of importance to ensure location information integrity, meaning that location information is original (from the generator), correct (not bogus or fabricated) and unmodified (value not changed). We present validation mechanisms to provide location integrity for VANETs. In the initial mechanism, we assume that all vehicles are equipped with a radar, a GPS receiver, and a transceiver. Since radar has a limited radar range and transceiver has a limited transmission range, we build network cells as a security unit as well as a communication unit. To ensure the intra-cell position information integrity, we propose an active validation mechanism (called active location integrity) that actively validates and enhances position security by enlisting the help of on-board radar to detect neighboring vehicles and to confirm their announced coordinates. Since radar is not currently installed in many vehicles, we weak the assumption by removing radar from the vehicle's equipments and propose the second mechanism (called passive location integrity) that maintains the mobility history records of vehicles, called the Map History. Based on a vehicle's Map History, we can predict a region where the vehicle will be present. The predicted region can be used to validate the announced position. In reality, vehicles are deployed with different combinations of equipment and some old vehicles may not have these devices. We address a validation mechanism (called general location integrity) which filtered and refined the location measurements obtained by the above active and passive location integrity methods. The three mechanisms above provide intra-cell position information integrity.

Since applications often involve position information of remote vehicles or entities which are beyond a cell (ranging to miles), we provide inter-cell position integrity as

well. Vehicles request that neighbors or opposite-side vehicles check the announced position information of remote vehicles. Both the request and response messages will be propagated among cells. Because of the high mobility of vehicles, the routing path is fragile. To improve location availability, we propose a stable routing scheme which will select and maintain stable routing paths. Both selection and maintenance of routing paths are based on a proposed probability analysis of VANET links. In addition, plaintext location information, especially aggregated location information, is vulnerable to attack as an attacker could easily modify the location information and harm the location integrity. We propose both encryption/decryption and access control mechanisms to provide location information confidentiality. The aggregated position message is encrypted by a key which is a geographic location which specifies a decryption region. Vehicles have to be physically present in the specified decryption region to decrypt or access the aggregated position information. As we can ensure the position information confidentiality, integrity, and availability, we achieve position information security based on the security requirements outlined in the CIA model (confidentiality, integrity, and availability).

ACKNOWLEDGEMENTS

Thanks and appreciation go out to all my committee members for being part of this journey and taking time to read this not only long but also detailed dissertation.

This dissertation would not have been successfully completed without the contributions of many people. My deepest appreciation and thanks to Dr. Stephan Olariu for his service as a inspirational supervisor, persistent direction, constructive discussion, and encourage throughout the program. My sincere thanks to Dr. Michele C. Weigle for her service as an co-advisor, valuable time, careful review, and constructive comments of my papers and dissertation. Special thanks to Dr. Jinhua Guo for being my external committee member and kind help in this project. Thanks to Dr. Kurt Maly and Dr. Ravi Mukkamala for being my committee members and their valuable feedback concerning this dissertation. I must also thank Dr. Irwin Levinstein for his efforts of signing me teaching commitments.

I would also like to thank my family for their support and encouragement. I will never forget my parents and my sisters' love, encouragement and expectation. My wife, Qi Li, deserves my warmest thanks and appreciations for her encouraging, sacrifices, and patience throughout the journey, and my son, David, for his lovely smile. You are the source of strong motivation to work hard to fulfill this goal.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The last few years have witnessed an unmistakable convergence of Vehicular Ad hoc Networks (VANETs) that promise to revolutionize the way we drive. Various car manufacturers, government agencies and standardization bodies have spawned national and international consortia devoted exclusively to VANETs. Examples include the Car-2-Car Communication Consortium [5], the Vehicle Safety Communications Consortium [6], and Honda's Advanced Safety Vehicle Program [7], among others. In addition, third party providers have already started to offer Internet access, distributed gaming, as well as other features of mobile entertainment.

There are ubiquitous safe and secure applications in VANETs. Most, if not all, applications rely on location information. If the location information is not secured, most applications will not function correctly. Therefore, we are motivated to propose specially designed mechanisms to improve location information security.

In this chapter, we will introduce the basics of the specially designed mechanisms. We start from advance vehicles, location and location applications. In addition, we will introduce the threat model, the trusted information and the untrusted in information. With this information, we propose three mechanisms to improve location confidentiality, location integrity, and location availability. The motivation and contributions of the thesis are also listed in this chapter.

I.1 VEHICULAR AD HOC NETWORK BASICS

I.1.1 Advanced Vehicle

New technology brings not only streamlined design but also new devices to vehicles. The new devices can extend vehicles' capabilities in computing, communication and sensing. The provision of on-board Global Positioning System (GPS) devices has revolutionized driving. Similarly, the recent introduction of short-range radar in some top-of-the-line models promises to reduce the number of fender-benders and other accidents. In fact, on-board radar is already used in advanced cruise control systems [8]. We base our vehicle model on the smart vehicle proposed by Hubaux *et al.* [9]. Hubaux includes an Event Data Recorder (EDR), a GPS receiver, and radar in his smart vehicle model. Specifically, vehicles represented in this dissertation are

assumed to be equipped with the following features (all sections in this dissertation will use these features unless specified):

- A wireless transceiver. The transceiver adopts the standard Dedicated Short Range Communications (DSRC) [10] for fast communications which are short-range wireless communication channels specifically designed for automotive use and a corresponding set of protocols and standards (IEEE 802.11p [11, 12]). The transmission range of DSRC is 300 meters. We assume that location information can be modified by attackers before being transmitted by this device. DSRC for intelligent transportation systems operates in the 5.9 GHz band (U.S.) or 5.8 GHz band (Japan, Europe).

- A GPS receiver. The GPS receiver will provide accurate location for a vehicle. The location information includes latitude, longitude, altitude, speed and heading information.

- Radar [13, 14], such as microwave, infrared or ultrasonic radar. We assume that the omni-directional radar can detect neighboring cars within line of sight in a radius of 150 meters. Some cruise control systems already use this kind of radar. Although the driver can visually confirm the objects detected by radar in some cases, our system does not require human intervention at all. Vehicular radar is only used active location integrity and general location integrity models.

- A unique ID. The ID device can be an electronic license plate [9], issued by a registration authority annually. It is an electronically tagged number plate which can identify a vehicle whether it is on the move or stationary. It is also a standard-looking number plate with an embedded tamperproof, active, RFID tag [15]. The tag is self-powered and independent of the vehicle's power systems. Once the number plate is fitted to the vehicle, the unique ID is installed. No further modifications are needed. We assume that the ID can be changed by attackers when they construct a location message. For example, an attacker can launch a Sybil attack (a vehicle pretends to be many fake vehicles) by using an electronic license plate test unit to generate fake IDs.

- A location key GeoLock black box. This device will convert a location into a secret key which will be used in geographic location based encryption algorithm discussed in Chapter V.

- A digital map. The digital map is an electronic map (e.g. GPS navigation Garmin digital map). Attackers cannot change other vehicles' digital map.
- A computer center. The computer center will provide data processing, computing and storage, etc.
- A virus checker. We will not discuss virus injection-based attacks.

In our model, vehicles with some of these devices (e.g. GPS, radar) are already in production. For example, Toyota has developed a Pre-Crash Safety system [14] which uses millimeter-wave radar to sense vehicles and obstacles on the road. Sensor Technologies and Systems developed forward looking vehicle radar [13], which can detect obstacles with a 150-meter range. Furthermore, GPS and a computing center are popular vehicle accessories today. Since these components are already being installed in vehicles, there is no additional cost required to deploy our location security techniques except a wireless transceiver.

Vehicles equipped with computing, communication and sensing capabilities will be organized into a ubiquitous and pervasive network, i.e., Vehicular Ad-hoc Networks (VANETs). VANETs can provide numerous services to travelers, mainly in improved driving safety and comfort to passengers. For example, an accident occurs in an intersection. An alert message will be generated by the infrastructure at the intersection. The alert message includes the location, the accident time, and the expected clearing time of the accident. All vehicles in the range of the transceiver will receive this alert. The alert message will be further propagated by some of the receivers on the basis of different routing protocols. All the vehicles received the alert message will be informed about the accident.

I.1.2 VANET Applications

Although the initial impetus of VANETs was safety, many other applications have been proposed or developed. Schoch *et al.* [1] partition the current VANET applications into four categories: active safety, public service, improving driving and entertainment, shown in Table I.1.2. For the active safety application, there are accident prevention applications (such as speed warning on curve or hill road, infrared visibility helper, blind spot helper) and accident warning applications (such as accident awareness warning). The public service applications include emergency reaction (e.g. fire truck localization) and support for authorities (e.g. stolen vehicle

tacking and finding). The improved driving applications include enhanced driving applications (e.g. the intelligent adaptive cruise control application and driving focus helper) and the traffic efficiency applications (e.g. intelligent traffic flow management, digital map update.) Applications such as Internet access anytime anywhere, multimedia play, and so on, can belong to entertainment applications. Therefore, the above discussion shows that the active safety applications and the improving driving applications can help the safety of driving. The public service applications and the entertainment applications can improve the comfort of driving.

Most, if not all, applications in VANETs need location information to function appropriately. For safety applications, vehicles need to know each other's location to avoid accidents. For public service applications, vehicles must be aware of the location of emergency vehicles to move aside for them. For entertainment applications, both the location of vehicles and the location of resources are needed to provide a high quality service. We now highlight some typical applications that rely on location information:

- *Incident management* is an application that manages operations and actions after an incident occurs. It includes incident detection, incident assistance, and traffic recovery from the incident. Incident detection discovers the exact incident location and is the basis of the rest of the incident management operations and actions.

- *Collision warning* is considered as the most important safety application. This application warns drivers about an impending collision by sensing and calculating a safety distance from obstacles, like other vehicles, buildings or anything that may cause a collision.

- *Vehicle tracking* is an application that allows car manufacturers, logistic companies and other trusted parties to remotely monitor a vehicle's location and movement. The location information is collected, confirmed, and transmitted to a central location server.

- *Emergency vehicle avoidance* is an application that alerts vehicle drivers to give way to emergency vehicles like fire engines, medical ambulances, police vehicles, military vehicles, etc. To avoid these vehicles, drivers have to know the location of these vehicles and their own location.

TABLE 1: VANET Applications [1]

Purposes	Situations	Application examples
I. Active safety	1. Dangerous road features	1. Speed warning (curve, hill), 2. tunnel, low bridge warning, 3. traffic lights violation warning
	2. Abnormal conditions	1. Road condition sensing and warning (wet, sand, icing, flooding), 2. visibility helper (infrared detection), 3. detour warning.
	3. Dangers of accidents	1. Intersection accident alert, 2. blind spot and lane change warning, 3. backing car warning, 4. front/rear collision warning, 5. parking helper, 6. bikes/pedestrians crossing warning, 7. highway merge alert, 7. left/right turn assistant
	4. Accident occurred	1. Accident awareness warning
II. Public service	1. Emergency reaction	1. Emergency vehicle approaching alert, 2. emergency vehicle location/route awareness
	2. Support for authorities	1. Electronic vehicle plate/status, 2. electronic driver license and identity, 3. stolen vehicle tracking and finding
III. Improved driving	1. Enhanced Driving	1. Intelligent adaptive cruise control, 2. driver focus detection/alert, 3. driver trainning/helper
	2. Traffic Efficiency	1. Broadcasting of accident, 2. intelligent traffic flow management, 3. advanced navigation system, 4. digital map update, 5. resource locator service (gas station, parking, hotel, food, etc.)
IV. Entertainment	1. Vehicle Maintenance	1. Vehicle self diagnostics, 2. tires check 3. repair notification
	2. Mobile Services	1. Internet access anytime anywhere, 2. music/movie share/watch, 3. short messaging
	3. E-Commerce	1. Advertisement in traffic, 2. car rental service, 3. parking reservation, 4. cargo tracking; 5. toll auto-collection,

- *Resource awareness* is an application that informs drivers of resources nearby like gas stations, parking lots, shopping centers, etc.
- *Accident insurance claims and accident law enforcement* are applications that need accurate location information to determine accident liability.

I.1.3 Location Information

In VANETs, the location in a message can be part of the message as shown in Figure 1.a. Multimedia entertainment (movies, music, etc.) applications are examples which use this type of message. The copyright of movie is restricted in a certain region. The movie can be encrypted and be played in a certain decryption region. We have to make sure the movie is not played beyond the decryption region. E-business applications for VANETs (online shopping, toll payment collection, etc.) can use this type message to enhance the security of messages. For example, credit card information can be strictly encrypted and decrypted in a certain region. This restriction will greatly enhance the security of credit card information. Alternatively, the whole content of a message can be composed of location information, shown in Figure 1.b. Military vehicle movement on the battlefield can use this type of message. The greedy location routing protocol for creating the VANET will use this type of message. The location information of all vehicles is aggregated. The aggregated location information can be strictly encrypted and decrypted in a specified decryption region to enhance the security of the location information. We also expect the decryption region can move along with the receiver vehicle.

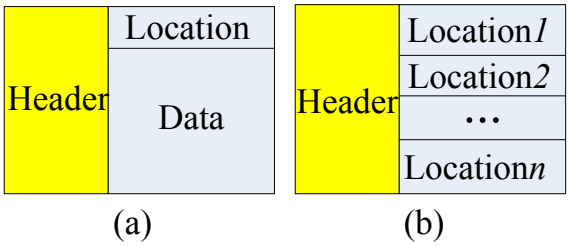


FIG. 1: Two types of location messages: (a) location is part of the message and (b) location is the whole content of the message.

Location information can be obtained from several devices, including GPS receiver, infrared scanner, etc. Since GPS imposes some constraints such as lack of coverage in some environments or its weak robustness for some critical applications,

other positioning techniques have been proposed for the vehicular field, including cellular or WiFi localization, dead reckoning (by using last known last location information and velocity), ultrasound range sensors [16], and image/video localization [17]. Critical safety applications such as cooperative collision warning and incident management need highly accurate localization. Some comfort applications such as parking reservation may benefit as well because an accurate positioning system can define the zone of relevance more precisely. Other services, however, do not require highly accurate localization, such as peer to peer applications, email clients, etc.

GPS location information in a vehicle is computed from received signals obtained from satellites high above the Earth [18]. Each satellite constantly transmits messages which include time, orbital information, and other satellite orbits. After receiving the message by GPS receivers in vehicles, vehicles can obtain the synchronized time. Vehicles use trilateration to compute their own locations. The location is displayed as latitude, longitude, and altitude. In this thesis we only deal with a two-dimensional coordinate system (latitude and longitude). Theoretically, three satellites are sufficient to compute a location. Since a small amount of clock error will result a large location error (because the speed of light is large), in reality, a GPS receiver uses more than three satellites to compute more precise locations. Some GPS receivers also show derived information such as direction and speed, calculated from location changes.

Location information can also be computed by using radio signals, infrared sensing, ultrasound and camera vision triangulation. If we think of GPS location as absolute location, then this kind of location information is relative location. The common computation basis is trilateration as well. Figure 2 shows the idea of trilateration. Vehicle D communicates with vehicle A, B, C through a transceiver. Vehicle D receives radio signals from vehicles A, B, C. Therefore, D can compute the relative location (a distance and an angle) of A, B,or C based on the received signal strength.

I.2 TRUST MODEL

I.2.1 Potential Attacks on Location Information

Since location information is fundamental and important in vehicular wireless networks, adversaries may attack the location information to harm the system in the

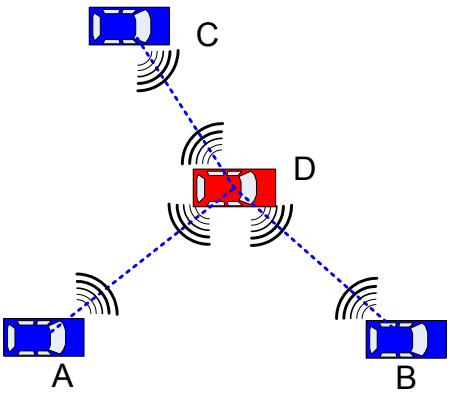


FIG. 2: Location computed by using trilateration.

following ways [19]:

- *Fabrication Attack.* Attackers create a bogus message, lie about the location of traffic congestion location, lie about identity, or lie about location. An attacker can fabricate its own location and announce the fake location to its neighbors and cell leader. An attacker (as a router) can fabricate other vehicle's location as a forwarded location packet.

- *Alteration Attack.* Attackers modify the location in the message. An attacker can modify its own location information. As a cell leader or router, an attacker can modify other vehicles' location information as well.

- *Packet Dropping.* Attackers serving as routers can simply drop packets. As a router, an attacker can drop packets directly to launch either a black-hole attack (dropping all packets) or a gray-hole attack (selectively dropping packets).

- *Replaying.* A malicious attacker pretends to be a vehicle in the past. The attacker re-injects previously received packets into the network. The attacker will pollute a node's location table by replaying beacons recorded in the past.

- *Geographic Sybil Attack* [19, 20]. The attacker pretends to be many other vehicles with fake location information. The attacker advertises multiple IDs and/or locations to mislead other nodes that high numbers of (non-existent) neighbors exist. Communication across non-existing nodes is in full control of the attacker; e.g., forwarded packets will be lost.

I.2.2 CIA model

Papadimitratos *et al.* [21] proposed security requirements for VANETs. These requirements are the minimum requirements of the NIST, information *C*onfidentiality, *I*ntegrity, and *A*vailability (CIA) model [22]. The CIA model is widely used in information security [23, 24]. The goal of the CIA model in information systems is to protect computers, software, and networks and the information they store, process and transmit. The CIA model in this dissertation means the following:

- Confidentiality refers to limiting information access and disclosure to only authorized users and preventing access by or disclosure to unauthorized individuals or systems.
- Integrity refers to the trustworthiness of location information. Information is original (from the generator), correct (not bogus or fabricated) and unmodified (value not changed).
- Availability refers, unsurprisingly, to the availability of location information, i.e. providing information whenever the information is needed. The availability of location information means that the protocol and service should be operational even in the presence of malicious or benign faults.

I.2.3 Trusted Information

We trust the following information:

- Radar detection. We trust the location which is "seen" by radar. Since radar detection has a measurement error which can be caused by shifting of the radar signal, we consider measurement error in angle error $\triangle\theta$ and radius error $\triangle\gamma$.
- Oncoming traffic. Vehicles in oncoming traffic can report a vehicle's location if the oncoming vehicle has the location information. The oncoming vehicle can obtain the location of a certain vehicle by using its radar. We trust this information because attackers in the oncoming traffic will pay an extremely high cost to launch a location attack. Even if there might be some attackers in the oncoming traffic, they will pass by in a short time and the damage will be minimal.

- Secret key from D-GeoLock. Since we assume this device is tamper-proof, the outputs of this device are trusted.

I.2.4 Untrusted Information

In this dissertation, the information that is protected is a tuple <time, ID, location>. We want to provide confidentiality, integrity and availability of this tuple. Before our security mechanism is applied, we assume that the attackers can change any value of <time, ID, location>. For example, an attacker can create a faked tuple about its own location. The attacker can modify time, ID or location value in other vehicles' location tuple. We also assume that the location message (e.g. the aggregated location message) is not signed by digital signature because we assume not all vehicles have Public-Key Infrastructure (PKI).

I.3 MOTIVATION AND OBJECTIVES

The original impetus for the interest in VANET was provided by the need to inform fellow drivers of actual or imminent road conditions, delays, congestion, hazardous driving conditions and other similar concerns. In time, however, it was recognized that the veracity of the traffic advisories is as important as the advisories themselves. A fabricated or doctored traffic advisory distributed by a malicious driver or bystander is apt to create slow-downs and even severe congestion. This simple fact of life has, in turn, spawned a substantial body of research in information security in VANET. Almost all advisories and other traffic-safety related messages depend in a critical way on location information. For example, traffic status reports, collision avoidance, emergency alerts, cooperative driving, resource availability, driver assistance, insurance justification of accidents, and accident rescue, etc., directly rely on location information. Online payment services, online shopping, and the like, mainly focus on network access. However, most of the time, these applications involve local services which need location information as well. Therefore accurate location information is of key importance. If location information is altered by malicious attackers, these applications will not work at all and will not be adopted by the public.

Therefore, it is of importance to ensure the security of location information. Yet, it has not been seriously addressed until recently. PKI is an important way to ensure information confidentiality [25]. We assume there is a trusted authority. The trusted

authority will generate keys for PKI. PKI has two keys, one is a public key which will be known by everyone and the other is a private key which is known only by the owner. The two keys are mathematically related each other, but the public key does not disclose the private key (within a long enough time). The PKI algorithm is known to all parties. If a public key is used by the PKI algorithm to encrypt information, a ciphertext will be generated. The ciphertext can only be decrypted by using the associated private key. The digital signature of a message often uses PKI to ensure the information integrity. In the following example, we illustrate how the digital signature works. A message M will be sent with a signed string $SHA(K_A^-(M), ID_A)$ where $K_A^-(M)$ is the encrypted message using private key of sender A and ID_A is the ID of the sender A. A digital signature can ensure the integrity of information if the receiver computes the signature and validates the received signature. When receiver B receives the message, B will compute the string $SHA(K_A^+(M), ID_A)$ where $K_A^+(M)$ is the encrypted message using public key of sender A. PKI is naturely adopted in VANET. For example, Hubaux *et al.* [9], Raya *et al.* [26], and Capkun *et al.* [27] apply PKI to encrypt information. There has been some research using digital signatures [28, 29, 30, 31] in VANETs as well.

On the other hand, PKI brings new challenges to VANETs. PKI includes both the public key and the private key. The public key needs to be known by other vehicles and the private key needs to be secretly stored. Therefore, key management is one of the challenges because of the large scale of the vehicle population. Some vehicles may have expired keys. To update the public and private keys, pervasive infrastructure will be required. Another challenge is that PKI requires homogenous configuration to achieve communication among vehicles. Some old vehicles may not have PKI installed. In addition, the overhead of PKI in terms of processing time will add significant processing time overhead to VANETs as well.

Several methods that try to provide location security for VANETs in the literature are presented. However, they all share the same shortcoming, namely that they cannot offer a comprehensive location security solution in VANETs because of the difficult network environment. In the literature, many information security algorithms claim to use PKI to achieve security. But information security by PKI encryption can only ensure confidentiality. PKI does not solve location integrity. A malicious vehicle could sign bad location information. There is no validation of the location information. For example in Figure 3, vehicle A and C can encrypt message

to generate a ciphertext, but those without the secret key can not access it. But, C can lie about its location, shown as C_1, C_2 or C_3. Vehicle A has no way to know about the lie. In addition, the ciphertext itself can be replayed.

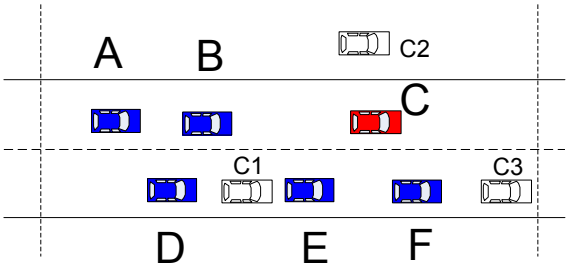


FIG. 3: Vehicle C lies about its location.

PKI does not solve location availability either. The main difficulty with location availability is caused by the high mobility of vehicles. Encryption of messages cannot help to select a reliable routing link and to disseminate message in a reliable way.

More importantly, in VANETs where multi-hop network connectivity is challenging, there is an ignored fact that location availability is fundamental. Location availability is a fundamental security feature of VANETs in which location information is always available when the location information is needed. Without this feature, location confidentiality and integrity have no value. Another ignored fact is that location confidentiality and location integrity do not provide location availability. A malicious vehicle can simply drop all of the messages that include location information. When location is needed most (e.g. in a collision avoidance application), location of vehicles is not available.

Therefore, our goal is to provide comprehensive location security protection for all applications in VANETs. Specifically, the objectives of this work are the following:

- achieve high location availability
- offer efficient location integrity
- provide feasible location confidentiality

Our goal of providing location *confidentiality*, *integrity* and *availability* satisfies the CIA information security model.

I.4 CONTRIBUTIONS

To ensure location information security in VANETs where the nodes have high mobility, we specially design mechanisms to ensure location confidentiality, location integrity and location availability. We call this the *location CIA* model.

Since the value of location information decays with time and distance, the location information of remote vehicles is often aggregated into a single packet. Because of the high mobility of vehicles, the routing path which delivers the aggregated location information packets is fragile. Location information that is not available when needed is almost as bad as none at all. It may be much worse for many applications like a traffic congestion notification application because drivers may be used to these applications and rely on them. Without location availability, these applications will not work when drivers need them most. To improve *location availability*, we propose a stable routing scheme in the second mechanism. The scheme selects and maintains stable wireless links which create a routing path. The selection and maintenance of wireless links are based on a probability analysis model of VANET links.

Location integrity means that information is original, correct (not fabricated) and unmodified (value not changed). We address a validation mechanism to provide *location integrity*. In the initial solution, we assume that all vehicles are enlisted with a radar, GPS, and transceiver. Since both radar and the transceiver have limited ranges, we build network cells as a security unit as well as a communication unit. To ensure the intra-cell integrity, our first proposed mechanism, active location integrity is proposed to validate location information by enlisting the help of on-board radar to detect neighboring vehicles and to confirm their announce location coordinates.

Since on-board radar is not currently installed in many vehicles, we propose a second solution *passive location integrity*, which does not need radar. The observer vehicle will send a location query of the observed vehicle and then will receive a set of locations of the observed vehicle from the neighboring vehicles of the observer vehicle. A statistical method is applied and the abnormal location values can be filtered out by this method. The correct location estimation can be obtained by averaging the remaining locations.

In reality, some vehicles have both on-board GPS and transceiver, some have transceiver only, and some old vehicles do not have any of these devices. Therefore, we propose the third validation solution to provide *general location integrity* under a real environment. The observer vehicle will collect a set of locations of the observed

vehicle from three data sources, radar detection, neighboring vehicles' reports and the on-coming vehicles' reports. The general location integrity solution can filter out the abnormal location values and obtain the validated location. These three solutions ensure intra-cell location information integrity.

Moreover, we ensure inter-cell location information integrity as well because numerous applications (for example, traffic notification, road view, etc.) involve location information of remote vehicles or entities which are beyond a cell (ranging to miles). Vehicles request that neighbors or on-coming vehicles check the announced location information. Both the request and response messages will be propagated among cells.

Given the insecure nature of wireless communication, the plaintext message is vulnerable because an attacker can easily modify the location information and corrupt location integrity. Therefore, we propose both encryption/decryption and access control mechanisms to provide *location confidentiality* in the third mechanism. In our approach, a special region (decryption region) is specified to the message receiver. The receiver must physically present in the decryption region to decrypt the received message. To achieve this idea, the receiver's location information is converted part of a key. We design an algorithm dynamic-GeoLock to convert the location information into the key.

The relationship of the modules addressed above is shown in Figure 4. Integrity prevents location information from alteration and fabrication. Confidentiality prevents location information from being exposed to unauthorized access. Availability means information is available when needed. The three components of location confidentiality, integrity, and availability, are interrelated. For example, confidentiality is often thought as information security by many people. However, confidentiality cannot detect fabricated information because an attack can create bogus information using the same encryption algorithm that regular vehicles use. Similarly, confidentiality does not improve information availability. As we can ensure the location information confidentiality, integrity, and availability, we achieve the location information security based on the security requirements outlined in the CIA model.

We summarize the main contributions of this dissertation as follows:

1. *Discovering link duration distribution*: In our analytical expression, we discover that the link duration has a log-normal distribution which is often assumed to be an exponential distribution in the literature.

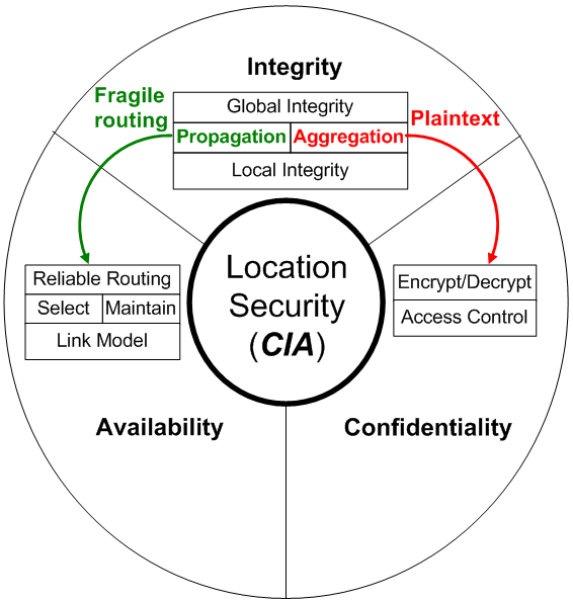


FIG. 4: Relationship of the proposed modules.

2. *Reducing control overhead in routing*: Based on our probability model and link duration time prediction, vehicles will send control messages only when the routing path is going to expire soon. This will save control messages and reduce control overhead.

3. *Reducing response time in routing*: Rebuilding a routing path is time consuming. We can reduce responding time because our location availability can maintain a more reliable routing path.

4. *First proposing active location integrity algorithm*: By using on-board radar, GPS location information can be actively validated by radar detection.

5. *Presenting a real world location integrity solution*: In normal traffic and most of scenarios, our location integrity solution can validate location integrity in a real world environment where some vehicles have GPS and transceiver, some have transceiver only, and some old vehicles do not have any of the proposed devices.

6. *Enabling location key dynamic computation*: Our D-Geolock algorithm can dynamically convert a location into a secret key instead of checking mapping tables.

7. *Ensuring location confidentiality*: Location messages are not exposed to one who are unauthorized to access them.

8. *Reducing control overhead in D-GeoEncryption*: Our D-GeoEncryption algorithm can send less frequent control message to update the decryption region.

9. *Improving location error tolerance in D-GeoEncryption*: Our D-Geolock algorithm can tolerate larger location errors than the original D-Geolock algorithm.

I.5 OUTLINE

The dissertation is organized as follows:

- Chapter II presents the state of the art in location security of VANETs. The related work includes security algorithms and solutions in mobile networks, wireless sensor networks, mobile ad hoc networks and vehicular ad hoc networks.

- Chapter III addresses the proposed module to improve location availability. A probability model is established by analytically deriving expressions of link duration and the probability of link duration, the basis of the proposed routing protocol. We then presented a specially designed routing protocol for propagating messages among vehicles.

- Chapter IV states the specially designed solutions to improve location integrity. This chapter initially addresses a homogeneous system in which all vehicles are installed with the same devices, such as radar, GPS, transceiver. We improve the initial homogeneous system by removing radar. We adopt statistical data fusion to filter false location data and to refine low resolution location data. To match real world scenarios, we further improve it by accepting a mix of devices: some vehicles can have all devices, some vehicles can have GPS and transceiver, and some vehicles can have only transceiver.

- Chapter V explains our extension of the geographic location based encryption mechanism to provide location confidentiality. We redesign the mapping function GeoLock which can convert a location into a secret key and propose a dynamic GeoLock (D-GeoLock) which can compute the secret key on the fly instead of looking up preinstalled secret key-table.

- Chapter VI shows the evaluation work of our proposed mechanisms. We compare the proposed location security schemes to related work. Each simulation begins by describing the simulation environment and how we obtain simulation results. The results are compared and analyzed.
- Chapter VII contains a summary and a list of future work.

CHAPTER II

RELATED WORK

The aim of this chapter is to discuss previous attempts to achieve location security in VANETs. We will give a brief overview of the major solutions. Since VANETs are related to Sensor Networks and Mobile Adhoc Networks (MANETs), we review location security solutions in these related fields. We will start with location availability solutions in Section II.1. Since location availability is ensured by reliable location delivery over VANETs, we address reliable communication by selecting predictable and endurable routing links in terms of link duration and the probability of link duration. Therefore, we introduce link probability models and reliable routing algorithms. Section II.2 will cover the solutions in location integrity. The solutions include digital signature, resource-based, and radio signal strength based algorithms. Section II.3 discusses location confidentiality solutions. A geographic location based encryption algorithm will be introduced.

II.1 LOCATION AVAILABILITY

Because of the high mobility of vehicles, the topology of VANETs is constantly changing. Therefore, the routing links and routing paths are inherently unstable. Unstable routing links and paths will cause a problem in location availability. Since routing paths are not reliable, location information is not guaranteed to be available when it is needed. To provide a reliable routing path, there are several routing schemes in the literature. But these schemes share a common shortcoming in that they cannot accurately predict the link duration and link probability. The link duration is the expected time that the wireless connection can last. The link probability is the probability of the existence of a link. Therefore, there is no guarantee of the availability of location information. In this dissertation, we propose a reliable routing algorithm to guarantee location availability. Our algorithm computes link duration and link probability on the basis of vehicle mobility information such as distance, velocity, acceleration, and direction.

II.1.1 An Overview

VANETs have several properties that can be used in routing protocols. They are connectivity, mobility, infrastructure, geographic location, and the probability of certain events such as link existence and link duration. We can classify the routing protocols in literature based on these properties. Connectivity among vehicles can be achieved by sending packets among vehicles, i.e. flooding messages to all nodes. The flooding-based protocols are first proposed in MANETs and Wireless Sensor Networks (WSNs). Some flooding-based protocols proposed originally in MANETs and WSNs, such as AODV [32], DSR [33], and DSDV [34], have been extended to VANETs. Some protocols in VANETs, such as Biswas [35], Murthy [36], Abedi [37], and DisjLi [38], are proposed on the basis of flooding as well. We give an introduction of each type of routing method, as shown in Figure 5.

Mobility includes relative distance, relative speed, and relative acceleration, even including the direction or moving patterns defined by maps. Mobility is a unique property, compared with fixed networks like Ethernet and ATM. Even compared with MANETs, vehicles in VANETs have a larger mobility scale. They can turn right/left, accelerate, decelerate, and stop. Vehicles are also constrained by roads. This mobility information, therefore, can be used to predict the lifetime/duration of the routing path. PBR [39], DisjLi [38], Taleb [40], Abedi [37], Wedde [41] and NiuDe [42] adjust mobility parameters to route messages.

In some proposals for VANETs, infrastructure, such as Road Side Units (RSU), bridges, buildings, cellular base stations and even routine buses, is used. The infrastructure helps the robustness and security of VANET communication. Therefore, routing protocols, such as DRR [43], SARC [44] and Bus [45], adopt the infrastructure to propagate messages.

A GPS receiver is a useful device in modern vehicles. VANETs can use GPS location coordinates to locate other vehicles and to guide vehicles to trip destinations (addresses, shops, hotels, etc.). Therefore, GPS geographic location coordinates can be used to construct an efficient routing path. There are some routing protocols, for example, CarNet [46], Kato [47], Zone [48], Greedy [49, 50], ROVER [51] and LORA-DCBF [52], that follow the same idea: find the next router which is geographically closest to the destination vehicle.

Geographic location-related routing is based on the current location information without considering the dynamics caused by the high mobility of vehicles. Probability

theory is often used in dynamic systems to describe the likelihood of certain events, for example the probability of link breakage with a certain transmission power or a certain mobility parameter. Protocols such as Yan [53], GVGrid [54], CAR [42], DeReQ [55], and REAR [56] use probability to build the routing protocols.

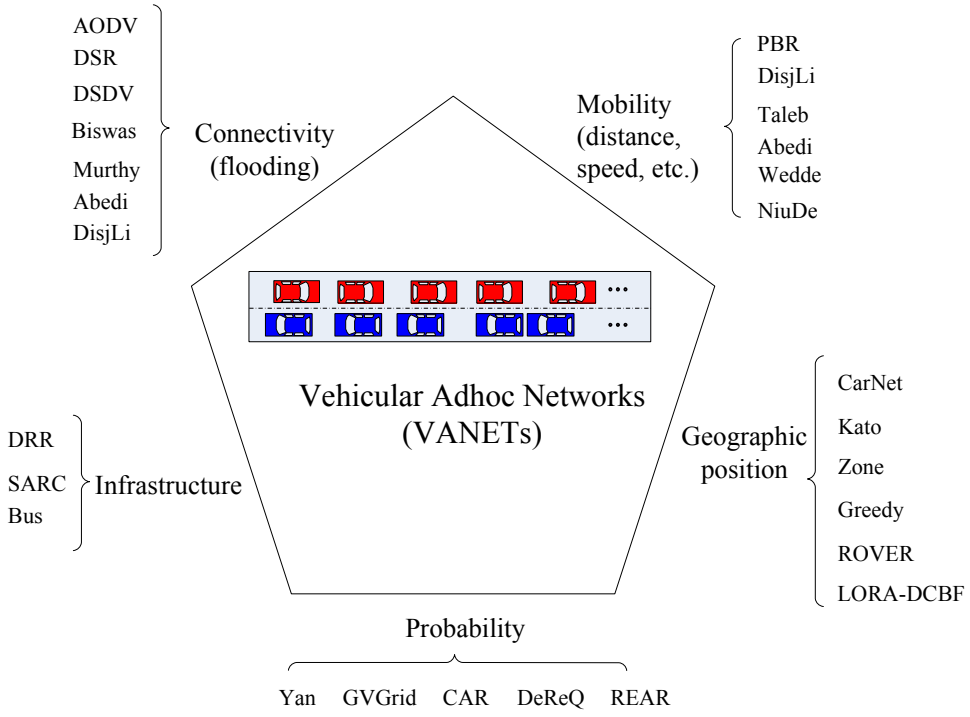


FIG. 5: An overview of reliable routing protocols in VANETs.

II.1.2 Flooding-Based Routing

Flooding-based routing is simple and easy to implement. This protocol is very efficient for broadcasting applications, for example traffic accident alerting applications. Figure 6 shows the basic idea of the flooding protocol. The source vehicle broadcasts a message. All of its neighbors will rebroadcast the received message. The rebroadcasting continues until the destination vehicle receives the message. During message propagation, if the message had been received before at one vehicle, the vehicle will drop the duplicate message. This protocol has some drawbacks. It is not efficient and has the potential risk to create a broadcast storm. Indeed, a broadcast storm may happen when the traffic density is high. The transmission attempts from a large population will cause even more responses and the responses will result in more

retransmissions in a snowball effect. The broadcast storm will waste the bandwidth and even block network communication.

For applications that need to unicast messages, the flooding protocol can be improved by dividing the packets into two types: control packets and data packets. The control packets are often RREQ (Route Request), RERR (Route Error), and RREP (Route Reply). The RREQ spreads out the routing requests among vehicles. The RREP returns the selected routing path along all the involved vehicles. Both RREQ and RREP are shown in Figure 6. The RERR reports the error encountered during the routing path exploration.

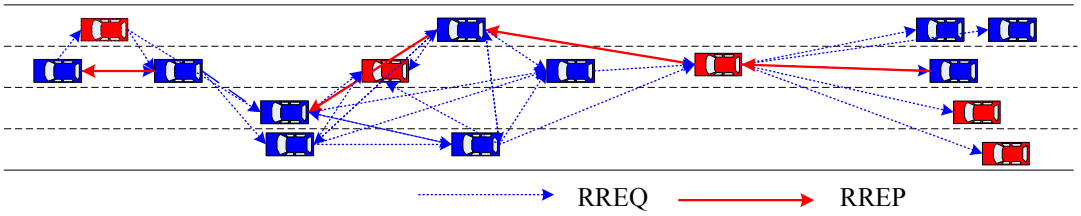


FIG. 6: Flooding routing protocols (the figure is not proportionally drawn).

In flooding routing protocols, the basic idea is to broadcast packets to the whole network. Each node receiving a packet will rebroadcast it if the node is not the destination node. It is not an efficient routing protocol in terms of bandwidth and delay. If all nodes can be reached within the transmission range of the sources's transceiver, the flooding method can be seen as an efficient routing scheme. However, in case of multi-hop communication, the performance of network will dramatically drop when the population of nodes increases. Flooding methods will generate a lot of duplicate packets, even causing a broadcast storm. In addition, the flooding methods scale badly beyond a few hundred nodes. But flooding is a reliable routing protocol in terms of availability, especially when the topology of the network is constantly changing and the traffic density is not high.

Since the pure flooding methods are costly in delay and bandwidth, enhanced flooding methods are proposed. The basic idea of the enhancement is to find a feasible routing path by broadcasting a control message, or probe, and then to send packets through the found routing path. For example, Murthy *et al.* [36], marked as *Murthy* in Figure 5, presented a wireless routing protocol by flooding control messages over the network which is viewed as a directed graph. AODV [32], initially

proposed for MANET, is often extended in VANET. It is a unicast on-demand routing protocol which includes two phases, route discovery and route maintenance. Four types of control packets are used: HELLO (a Hello message), RREQ, RREP, and RERR. Abedi *et al.* [37] (marked as *Abedi* in Figure 5) and Li *et al.* [38] (marked as *DisjLi* in Figure 5) extend AODV and use RREQ to explore the routing path as well. Additionally, there are several other flooding-based mechanisms used in MANET, for example DSR [33], and DSDV [34].

Biswas *et al.* [35], marked as *Biswas* in Figure 5, addressed a flooding routing method which extends the original flooding method by acknowledging the flooding message. When a node receives a packet, it rebroadcasts the packet. This vehicle watches the packets from behind vehicles. If it receives the same message from behind, it infers that at least one vehicle in the back has received the packet and that vehicle will retransmit the packet. Therefore, the event of receiving the same packet from behind vehicles is treated as an acknowledgment of flooding. If the vehicle does not receive the acknowledgment, it will periodically rebroadcast the packet until the acknowledgment is received.

II.1.3 Mobility-Based Routing

Mobility is one of the major differences between VANET and other network systems, even MANET. In wired networks, such as Ethernet and ATM, nodes are fixed in location. In conventional wireless networks, such as MANETs, which is often used in a small region like an airport, nodes often have slow mobility. In cellular networks, nodes can have fast mobility but the communications among nodes are often through infrastructure, i.e. base station. Nodes in VANET often have high mobility, for example fast speed and frequently changing speed and direction. The high mobility makes many of the existing conventional routing algorithms not applicable to VANET. Mobility, therefore, is used as a key factor to select and maintain a routing path.

Namboodiri *et al.* [39], marked as *PBR* in Figure 5, present a predictable mobility pattern of vehicles on highways and use it to create a route by predicting the lifetime of route and selecting the best route. Li *et al.* [38] (marked as *DisjLi*) and Taleb *et al.* [40], marked as *Taleb* in Figure 5, present a routing method by grouping vehicles according to velocity. The basic idea of the method is to predict a possible link breakage event prior to its occurrence by computing from vehicle's speed. Vehicles

are grouped into four different groups based on their velocity vectors (speed with directions). If the directions of speeds of two vehicles are the same, the link between the vehicles will stay longer than the link composed by two vehicles with different speed directions. The process of routing path searching is the following. Initially, the source node broadcasts a request packet, RREQ. This RREQ will be disseminated among nodes by rebroadcasting. The most suitable path is chosen when the RREQ is received by the destination. The duration of the path is predicted by the vehicle's speed and the distance of the two vehicles. A new route discovery is always initiated prior duration of the routing path, i.e. the shortest link duration.

Abedi *et al.* [37] (marked as *Abedi* in Figure 5) present a enhanced routing protocol based on AODV to adapt to the high mobility of vehicles. The protocol uses three mobility parameters: location, direction and speed. This method treats direction as the most important parameter to select the next hop because the nodes moving in the same direction will be more stable than nodes in the opposite direction. Therefore, this method will select routing links composed of nodes in the same direction as the source and/or destination nodes. Moreover, location is the second most important parameter that is used for next hop selection.

Wedde *et al.* [41] (marked as *Wedde* in Figure 5) address a routing algorithm based on a rating value . The rating value is computed to evaluate the road conditions (actual traffic situation) based on the interdependencies of average vehicle speed, traffic density and the traffic quality (in terms of congestion). A routing link is incorporated into a routing path if the rating value satisfies a certain requirement, i.e. the rating value. This method only uses part of mobility information, i.e. average vehicle speed, which is not sufficient to compute the path accurately. In addition, the average vehicle speed is not the reason for link breakage, but rather it is the relative speed between the sender vehicle and the receiver vehicle.

Niu *et al.* [42] (marked as *NiuDe* in Figure 5) propose a new link reliability mathematical model which considers not only the impact of the link duration but also the traffic density. A purpose of this model is to find a route which is not only reliable but also compliant with delay requirements in multimedia application. This link duration is computed by using two mobility parameters. The distance between the sender vehicle and the receiver vehicle is divided by the relative speed of the sender vehicle and the receiver vehicle. The accelerations of the sender vehicle and the receiver vehicle are not considered in the computation. But the accelerations will

greatly affect the link duration.

II.1.4 Infrastructure-Based Routing

In hybrid VANETs, stationary roadside units (RSUs) are combined with the on-board units (OBUs) equipped on vehicles to provide reliable routing and differentiated applications. RSUs act as fixed reliable nodes. They are connected by backbone links with high bandwidth, low delay, and low bit error rates. Vehicles can directly communicate with each other through wireless links. Figure 7 shows the infrastructure. When the link between two vehicles is broken, the RSU will act as a node to relay packets to the destination vehicle. This routing protocol is the most reliable and feasible in reality. However, the drawback is that the infrastructure is costly and limited to urban areas. In the scenario of rural areas, especially during disasters like hurricanes and earthquakes, infrastructure will be damaged first, and the traffic information will not be able to be delivered to the drivers when they need it most.

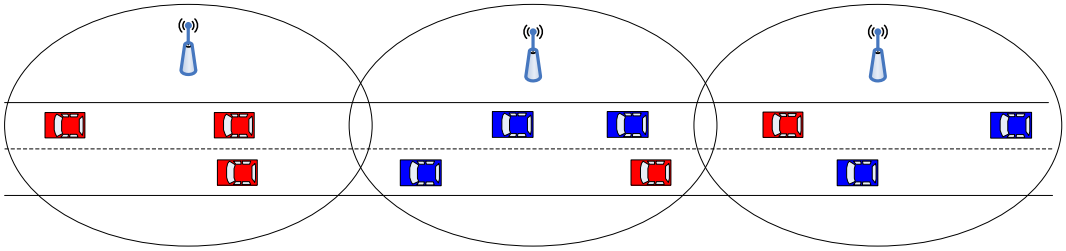


FIG. 7: Stationary roadside units can be used in hybrid VANETs.

He *et al.* [43] (marked as *DRR* in Figure 5) propose two notions, virtual equivalent node (VEN) and differentiated reliable path (DRP) to protect against link failures. If the routing path is broken, RSUs (one or multiple) will act as a VEN to provide connection, for the broken clusters of vehicles. After a vehicle successfully connects with an RSU, its location information is synchronized to all related RSUs instantly.

Kim *et al.* [44] address a novel routing protocol, called *SARC*, which can find the routing path and protect privacy in a route discovery and data forwarding phase. The method adopts a street-based path calculation algorithm for route discovery. The identity, location, and route anonymity are defined and analyzed as well.

Kitani *et al.* [45] (marked as *Bus* in Figure 5) propose a information routing method that uses buses traveling along regular routes as message ferries. Buses are

an ideal medium to avoid losing traffic information statistics of areas that buses travel. Each vehicle measures time consumption to pass an area. Traffic information statistics can be computed from the information received by cars in inter-vehicle communication.

II.1.5 Geographic Location Based Routing

Location based routing protocols have received a great deal of attention in VANETs. These routing protocols are very efficient in high mobility ad hoc networks since the path to a destination can be determined by the location information of neighbor and destination nodes. The fundamental idea of geographic location based routing protocol is shown in Figure 8. The geographic location values of vehicles are used to partition the nodes into sub-sections of roads. The sub-sections, or zones, can be dynamically created on the fly and can be predefined and installed on each vehicle. Therefore, vehicles in a zone become members in a group. Each group only has one or two vehicles functioning as gateways to relay packets. Other members in the group keep silent and drop the packets. The advantage of this method is to reduce the duplicates of packets and therefore, to improve the delay and bandwidth utilization. But this method, especially for the routing method based on dynamic group creation, will introduce other issues like group management, as well as overhead of control messages because vehicles have to know their neighbor location information. Another drawback is that this protocol may not find the optimal routing path because relative mobility is not considered.

Bachir *et al.* [57] and Ibrahim *et al.* [58] address a geographic location based routing protocol. The protocol uses the geographic location as a parameter to select the next router, i.e. Inter-Vehicle Geocast (IVG). Ibrahim *et al.* [59] improved one shortcoming in the protocol proposed by Bachir. If the traffic is dense, there may be a broadcast storm. The broadcast storm can be avoided by a probability. The probability of sending message is a random number $(0, \frac{1}{density})$ where $density$ is the density of the traffic.

CarNet [46] uses a grid to propagate packets. The grid is defined on the basis of the geographic location of nodes. A forwarding method based on geographic location and a scalable dissemination service based on location are designed to route packets from car to car without flooding the whole network. CarNet, as announced by the authors, can support many applications: IP connectivity, cooperative highway

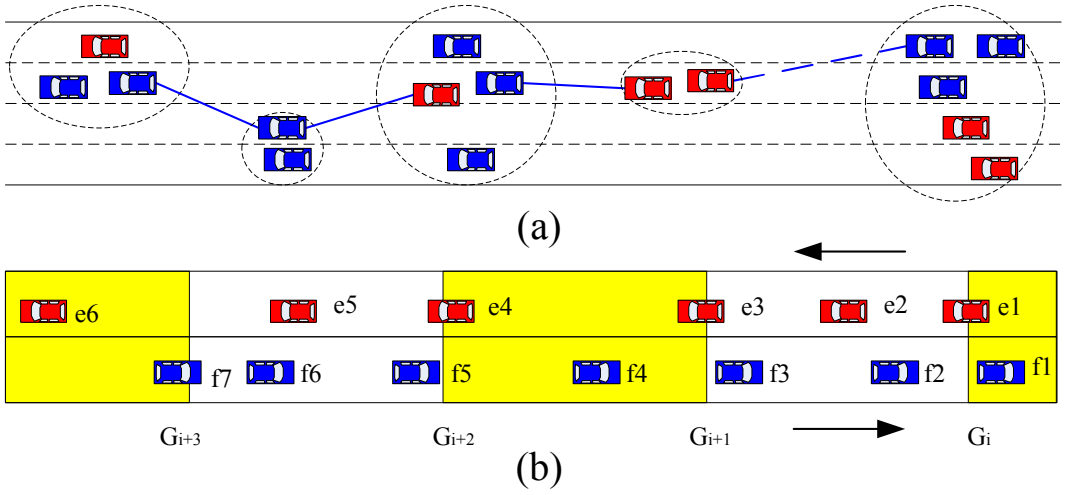


FIG. 8: Geographic location-based routing protocols. The circles and shaded area represent zones

congestion monitoring, fleet tracking, and discovery of nearby points of interest. Kato *et al.* [47] (marked as *Kato* in Figure 5) propose a method for constructing network groups according to lane location and evaluate the proposed method by simulation as well.

Bronsted *et al.* [48] (marked as *Zone* in Figure 5) present a zone flooding algorithm and a zone routing algorithm. A zone is defined as a geographic area, for example 500 meters of a road. If vehicles are in the zone, it allows them to broadcast packets. Otherwise the received packets will be dropped. The effect of the zone is that packets are only delivered in a particular section of a road.

Gong *et al.* [49] and Lochert *et al.* [50] present a greedy routing method (marked as *Greedy* in Figure 5) on the basis of geographic positions of vehicles, as shown in Figure 8. Each vehicle knows its own geographic location by the help of an enlisted GPS receiver or some other method, for example radio signal strength. Vehicles rebroadcast packets only at the nodes whose distance to the sender is the longest. The fact that vehicles transmit packets aggressively toward the destination gives the name "greedy". The directions of vehicles' movement are included in the geographic greedy routing. The direction of movement helps to select long duration links. But the mere direction of movement does not help much. The mobility of vehicles are

considered and predictive directional greedy routing is proposed [49].

Kihl *et al.* [51] describe a routing protocol, called *ROVER* (RObust VEhicular Routing). Zones are defined on the basis of positions of vehicles which are equipped with GPS receivers. ROVER broadcasts a control packet, similar to AODV, among zones to find a routing path. Once the routing path is found, data packets are unicasted by the routing path to improve the performance of communication. Similarly, Momeni *et al.* [52] propose a reactive algorithm for mobile wireless ad-hoc networks, called Location Routing Algorithm with Directional Cluster-Based Flooding (LORA-DCBF). LORA-DCBF uses a zone-like group routing method. To reduce the number of Location Request (LREQ) packets and minimize duplicate retransmissions in the same region, the control traffic packets are selectively transmitted by selected nodes, called gateways. All the members in the zone can read and process the packet, but do not retransmit the packets. Only gateway nodes retransmit packets between zones, i.e. gateway to gateway communication.

II.1.6 Probability-Based Routing

Because of high mobility, the topology of a VANET is changing constantly. Probability theory is an ideal tool to find a routing path, for example the probability that a wireless link exists between two nodes at a certain time, or the probability that a wireless link will stay connected for a certain time interval. In this section, we will give a short survey on probability-based routing protocols.

Jiang *et al.* [56] present a routing method, called *REAR*, based on the receipt probability of alarm messages of nodes. REAR can compute the receipt probability of alarm packets based on the real wireless channel in VANET. The selection of the next hop is based on the receipt probability. The probability model is based on wireless signal strength and the loss of signal. The wireless signal loss is composed of two parts, the path loss and the diffraction loss. The receipt probability is computed by using the relationship between packet loss rate and received signal strength. The receipt probability values of all neighboring nodes are estimated from the received signal strengths. The routing path with highest receipt probability value will be selected as the routing path.

Niu *et al.* [42] address a routing protocol (marked as *DeReQ* in Figure 5) that dynamically creates and maintains a robust route to provide QoS for multimedia applications over VANETs. The routing methods use parameters, reliability and

delay. The reliability is on the basis of a probability function that can predict the future status of a wireless link. The probability function is introduced in the literature [60, 61] and is defined as the probability that there is an active link between two nodes. The digital map and the GPS device are used to find the route with best reliability. The route is maintained by proactive communication among intermediate nodes. If a link is going to break, the routing path will be rebuilt before the link breaks.

Yang *et al.* [55] develop a new routing protocol called connectivity aware routing (*CAR* in Figure 5). The basis of this method is the connectivity probability model of each road segment. A route with the highest probability of connectivity to forward packets will be selected. As claimed by Yang, the packet delivery ratio using the proposed method can be increased up to at least 90% and the delay is in the acceptable range as well. The connectivity model is on the probability computation on a road which is partitioned into grids/cells. The unit of cell is the average length of a car, 5 meters. Then the probability to compute the connection between two nodes is to compute the probability that the distance between the two nodes is within a certain value (transmission range of wireless).

Sun *et al.* [54] propose an algorithm (marked as *GVGrid* in Figure 5) to find a reliable routing path that was compliant with delay requirements. The algorithm is based on several assumptions, namely (1) intermediate nodes are equally spaced, and (2) vehicle speed is normally distributed. Based on these assumptions, they compute the probability of link lifetime as the reliability of a link. By querying possible links or paths, a path with high reliability and sufficiently small link delay will be selected as the routing path. However, assumption (1) is not reasonable since, as known, the inter-vehicle distance is a random variable and certainly not constant.

Our previous work addresses a ticket-based routing method [53] (marked as *Yan* in Figure 5) in VANET by using the expected duration of links. The expected link duration is computed by a probability model. The Ticket-Based Probing with Stability conStrained(TBP-SS) Routing method is also proposed [53]. The key parameter used to select a route is based on the mean duration of a link (defined as stability) which is computed by the probability model. From the "divide and conquer" algorithm, each optimal routing link is selected and results an optimized routing path which is composed by each optimized link. Vehicles are equipped with a DSRC/IEEE 802-based wireless transceiver (e.g. IEEE 802.11p) and a GPS device.

Cheng *et al.* [62] proposed a method on the assumption of a *Poisson* process. The distance of a broadcast message can be propagated in a network which contains homogeneous *Poisson* distributed nodes. Several bounded areas (e.g. a straight line, a circle, etc.) are studied. Piret *et al.* [63] addressed the connectivity issues on a one-dimensional line segment where nodes are uniformly distributed. There are other routing algorithms on the assumption that vehicles are uniformly distributed [64, 65, 66]. In VANETs, vehicles move on roads. But roads can be circles, rectangles, straight lines, etc. It is not sufficient to limit the road as a certain bounded shape, like a straight line, a circle, or as one-dimensional shapes.

Bettstetter *et al.* give a tight lower bound of connectivity of multihop radio networks for the minimum node density [67]. The basic assumptions are (1) the wireless network is in a log-normal shadow fading environment, (2) nodes have equal transmission capabilities, (3) the number of nodes at any time is randomly distributed according to a homogeneous *Poisson* process. Based on the three assumptions, the probability of the link and the node degree are derived. Bettstetter *et al.* can be used in sensor networks and MANETs. However, in VANETs, the mobility of vehicles can greatly change the topology of the network. For example, a network with high speed nodes is less likely to have a stable link. But the velocity of nodes is not shown in the probability of a link. In our model, we analyze the probability model based on the mobility and a log-normal shadow fading environment.

Nekovee *et al.* [68] assumed that the distribution of car velocity is a normal distribution. The path loss is also formalized as an exponential function of velocity. Nekovee *et al.* [69] proposed a model to compute the probability of a link in VANETs. The distance headway is expressed by a constant mean speed times the time. The distance headway computation does not include the relative speed and acceleration between the sender vehicle and the receiver vehicle. This paper only considers the slow fading/shading radio propagation model. Moreover, this work only considered the radio signal without considering the mobility of vehicles.

II.2 LOCATION INTEGRITY

Location information in this dissertation refers to a tuple <time, ID, location>. Due to the limit of transmission range specified by DSRC (300m), the communication connection among vehicles is in a multi-hop mode, i.e. a location message will be

propagated by several intermediate nodes before the message arrives at the destination. There is a risk that the content of the location information can be changed or fabricated by malicious attackers. Therefore, a validation mechanism needs to be addressed to improve location integrity.

Public Key Infrastructure (PKI) [9, 26, 27] and digital signatures [28, 29, 30, 31] have been used for location integrity. But there are some shortcomings. PKI-based encryption solutions add significant overhead to the system in terms of processing time and response time. More importantly, message encryption does not solve location integrity because a message can be fabricated by the sender or be replayed by intermediate nodes. Although a digital signature can provide location integrity by signing the message, the key management (distribution and updating) of the secret keys or public/private key pair is challenging because of the large scale of vehicle population. In this dissertation, we take a different approach. We allow vehicles to send the information in plain-text and depend upon receivers to verify the information.

Leinmüller *et al.* [70] proposed a method to secure location information by using hard thresholds to detect false locations. Vehicles monitor data to verify the reported location. If the reported location lies beyond a threshold, the location is determined to be false. If the number of nodes is larger than a threshold maximum number of nodes, the honest nodes know that there are some fake nodes. Although the authors do not use any other devices or hardware, the accuracy and efficiency are difficult to guarantee. In addition, this method is not flexible because of the high mobility of vehicles in VANETs.

Radio signal-based methods [71, 72] can determine false claimed positions based on the received signal power. The basic idea of this method is that the distance between two nodes can be computed from the received signal power. If there is a vehicle at a location which does not match the distance computed from the received radio power, the location of the vehicle is determined to be a fake. However, a malicious node can use the same method to compute the transmission signal power to fool other nodes. Also, the radio signal may bounce off of vehicles and other obstacles. Detection based on these bounced radio signals may not be accurate.

Resource-based methods test vehicles' resources, such as radio resource [73], computational resources (vehicles failing to solve a puzzle are identified as fakes) [20], and identification resources (vehicles whose MAC and IP addresses which are not recorded in a profile are identified as fakes) [74]. Newsome *et al.* [73] claimed that

the method for detecting Sybil attacks proposed by Douceur [20] is not applicable to ad hoc networks and proposed other prevention methods including radio resource testing, central registration, and location verification. Radio resource testing is based on the assumption that no device can send and receive on more than one channel at a time. But, the attackers may have multiple channels. Central registration requires all vehicles to authenticate themselves before sending messages. Central registration may not work because the attackers can simply create multiple identities, like Sybil attacks. Besides, registration creates a privacy concern. Piro *et al.* [74] records vehicles' MAC and IP addresses as a passive ID to create a profile about neighbors. The profile can be used to validate the neighboring vehicles' location. However, attackers can have multiple devices to defeat this method. Moreover, privacy is an issue if MAC and IP addresses are recorded and tracked.

In this section, we stated location integrity solutions in the literature. There are PKI-based location encryption methods, digital signature solutions, radio signal based solutions, and resource-based methods. But there are some shortcomings in these solutions as we addressed above. We specially design a light-weight and efficient location integrity solution. We start with a strong assumption that all vehicles are equipped with radar, GPS receiver, and transceiver. When a receiver receives a location announcement, the receiver will use its radar to detect the location of the announcement. Then we weaken the assumption by removing the requirement of radar. A receiver will consult its neighboring vehicles about the location announcement. The receiver will apply a statistical method to filter and refine location information to obtain an agreement of location. If the announced location is very close the agreed location, the announced location is validated. We further weaken the assumption by assuming that some vehicles have different level of combinations of the devices. We propose a real-world solution. A receiver will collect location information from three different resources: radar, neighbors, oncoming vehicles. We apply a statistical method to obtain the location estimation.

II.3 LOCATION CONFIDENTIALITY

In a wireless environment, location in plaintext mode is vulnerable to attackers. For efficiency, the message which includes location information can be in plain-text mode to transmit. Wireless media is open to the public, and everyone in the wireless network can access the content of the message. Therefore, the content of the message

will be exposed to unauthorized access. This will harm location confidentiality. In literature, there are several attempts to enhance information confidentiality. We will give a brief overview of the major solutions.

II.3.1 Encryption and Authentication

There are two basic types of encryption algorithms, asymmetric and symmetric. In asymmetric algorithms, each node has a public key and a private key. The public and private keys are special in that a message encrypted with a node's public key can only be decrypted using the node's private key, and vice versa. In public key infrastructure (PKI), a well-known mechanism for using and distributing public keys, a Certification Authority (CA) is responsible for validating public keys and distributing certificates used for authentication. In symmetric algorithms, the communicating peers share a secret key. Both encryption and decryption are performed using the same secret key, thus the secret key must be protected.

PKI and digital signatures are well-explored methods in VANETs [29, 9, 75]. A CA generates public and private keys for nodes. When a node A sends an encrypted message M to node B, A will encrypt M by using the public key of B. Only B has the private key, so only B can decrypt the ciphertext. If B wants to digitally sign the message M, B encrypts the message with its private key and sends both M and the signed version of M to A. A then verifies the signature by using B's public key to decrypt the signed version. If the result is M, A will accept M as sent by B, because only B has the private key to generate the unique signature. Laberteaux *et al.* [76] discussed applying a similar method to sign messages in VANETs. The purpose of the digital signature is to validate and authenticate the sender. The purpose of encryption is to disclose the content of message only to the nodes with the secret keys. PKI is a method well-suited for security purposes, especially for roadside infrastructure, like roadside e-shops, Internet access points, etc.

But, there are some issues in using PKI in VANETs. The main problem is the need for a trusted CA to distribute public keys and certificates. In order for all vehicles to be able to communicate with each other, all vehicles will have to trust the same CA, a difficult requirement when vehicles are manufactured by different companies in different countries. In addition, bad or mis-used certificates must be revoked. The list of revoked certificates must then be distributed to all vehicles. Another problem is that asymmetric encryption/decryption often takes 1000 times

longer to perform than symmetric encryption/decryption [2]. In addition, nodes in VANETs can communicate in groups [77, 78, 26, 75]. In this case, the requirement of public keys for all nodes is not needed, because the vehicles in a group can share the message. In our work, we improve the encryption/decryption speed by using symmetric algorithms. Therefore, we have to design the secret key. The secret key is based on the location of vehicles and no extra cost needed. Moreover, the secret key is a group key which can be shared by a group of nodes.

II.3.2 Location-Based Encryption

Location-based encryption was proposed by Denning *et al.* [79, 2] to limit the area inside which the intended recipient can decrypt messages. This *GeoEncryption* integrates geographic and mobility information (such as location, time, speed, etc) into the encryption and decryption processes. Denning proposed GeoLock, which is computed with the recipient's location, velocity, and time, as shown in Figure 46. Suppose vehicle A wants to send a message to vehicle B. The GeoLock is a secret key converted from the location by a mapping table shown in Figure 10. The GeoLock of A is processed by modulo operation with a secret key Key_S and then the result is encrypted by the public key Key_E of B and sent to B which decrypts the ciphertext using the private key Key_D of B. The secret key (symmetric key) Key_S is obtained and used to decrypt the message.

Al-Fuqaha *et al.* [80] extended Denning's GeoEncryption model by providing decryption zone prediction under mobile networks. The decryption zone is a specified region where the encrypted message is allowed to be decrypted. However, the decryption zone by Al-Fuqaha is designed for slow or constant mobility nodes. The location predicted by Al-Fuqaha does not consider prediction errors. But in VANETs, the nodes have high mobility which will definitely cause a certain range of prediction error. Vehicles can move about 33 meters per second (75 miles/hour) and can turn at street intersections, stop, accelerate, decelerate, etc. The dynamics of a vehicle's mobility will make the prediction from a sender difficult and inaccurate. Therefore, Al-Fuqaha's extension is not designed for VANETs.

Reddy *et al.* [81, 82] proposed an application of Denning's GeoEncryption. The latitude/longitude coordinate of a message decrypter is used as the key for data encryption and a toleration distance (TD) is designed to overcome the inaccuracy and inconsistent problem of GPS receiver.

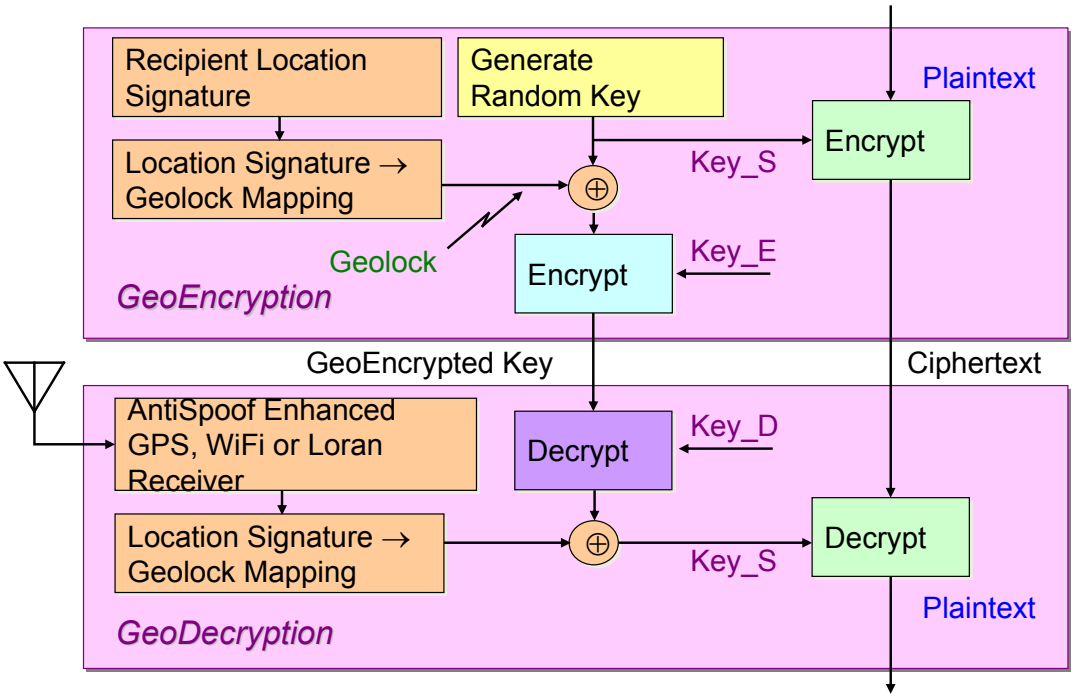


FIG. 9: Denning's GeoEncryption [2] (Used with permission.)

Zhang *et al.* [3] proposed a location-based security algorithm in wireless sensor networks (WSN). Both ID and locations of nodes are hashed and served as a public key. Suppose Alice and Bob are nodes in WSN. Bob wants to send a message to Alice. They have a key handshaking process, shown in Figure 11. Bob will broadcast a "helloLBK" message in the network. Alice will unicast a reply including its ID, i.e. ID_A. Bob will send an encrypted message processed by a secret key IK_A which is an outcome of hashing ID_A. The encrypted message includes ID_A, the location of Alice and the hashed location LK_A. The LK_A can serve as a secret key. There are two issues in this algorithm. First, IDs are broadcasted over the whole network, so privacy is a potential problem. Second, this algorithm lacks location error tolerance. Location measurements often include precision errors. A hash function may produce different keys for Alice in different measurements.

Eschenauer *et al.* [83] proposed a security algorithm. The key contribution is key management. For each node, there is a set of keys which are randomly selected from a set of global keys. The requirement of the selection is to ensure that there exists a common key between any two nodes within a certain probability.

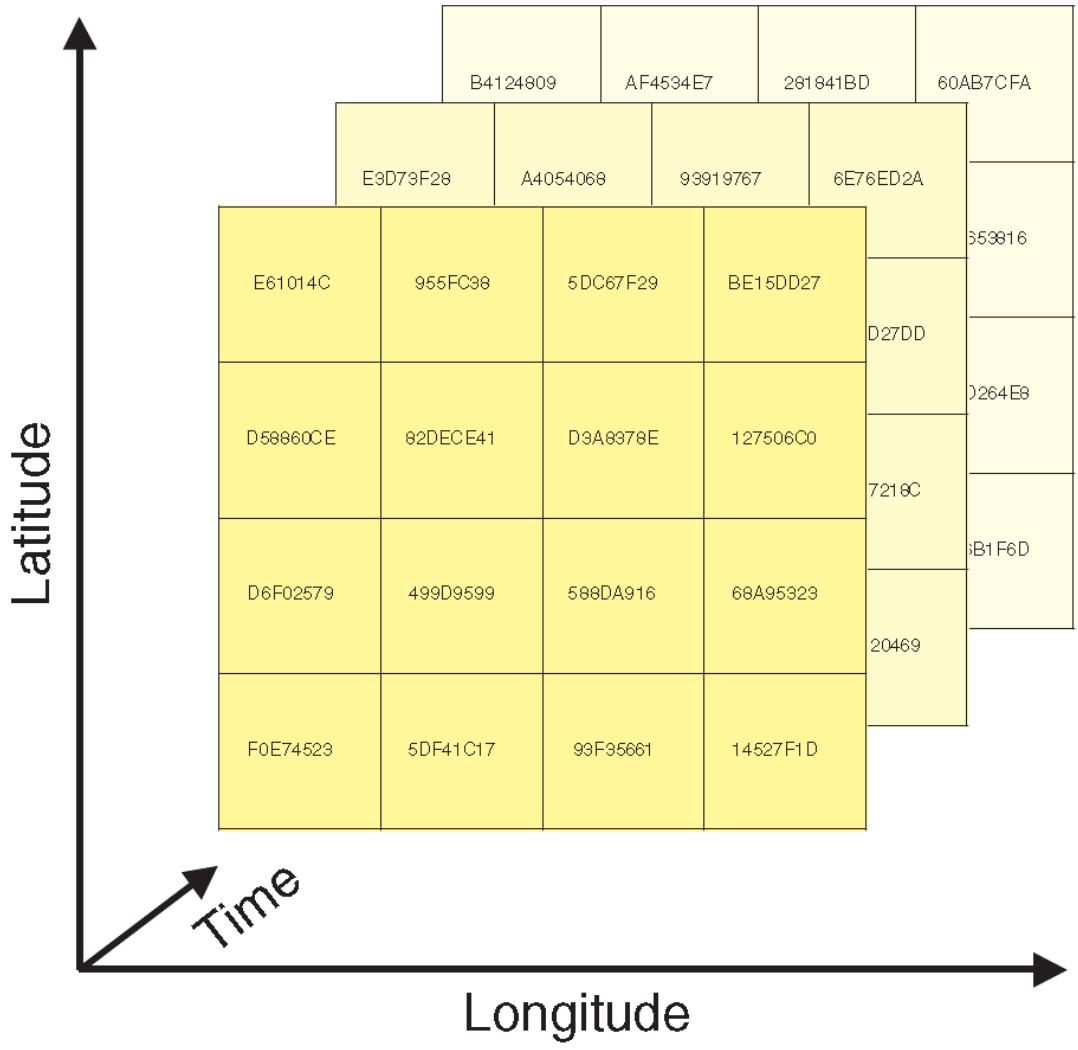


FIG. 10: Denning's GeoLock table [2] (Used with permission.)

$$\text{Bob} \longrightarrow \text{Alice} : \text{“helloLBK” (broadcast);}$$

$$\text{Bob} \longleftarrow \text{Alice} : ID_A \text{ (unicast);}$$

$$\text{Bob} \longrightarrow \text{Alice} : \{ID_A, pos_A, LK_A\}IK_A \text{ (unicast).}$$

FIG. 11: Location based security in wireless sensor network (WSN) [3]. Alice and Bob are nodes in WSN.

Cho *et al.* [84] proposed a location authentication and authorization algorithm which can authenticate the location announcements of mobile nodes. The algorithm relies on access points to distribute secret keys and to locate nodes. Although this algorithm does not require GPS, the accuracy of localization is not high especially in some environment such as among the high buildings or under tunnels.

In our work, we design the key composition/recovery in detail. No mapping tables are needed. Positions can be mapped to a lock on the fly. Since nodes in VANETs have high dynamics, the decryption region is designed as a series of fixed-size squares. The area of the square is sufficiently large to cover the error of the decryption region prediction. Moreover, we incorporate prediction error by using location prediction deviation. We trade freedom of the size and the shape of the decryption region in order to obtain the feasibility and the accuracy of decryption region prediction.

II.4 SUMMARY

In this section, we presented related work on enhancing location security in location availability, location integrity and location confidentiality. We have started the section by showing location availability solutions. Location availability can improved by more reliable routing protocols. Five types of routing protocols are reviewed. The link duration and link probability are used as parameters to construct and to maintain a routing path in our solution. Therefore, we also reviewed the probability model of a link in related fields, e.g. VANETs, sensor networks and mobile ad hoc networks. In location integrity solutions, we have showed digital signature methods, radio signal-based methods, and resource-based methods. Finally, we addressed location confidentiality solutions. Two types of confidentiality methods are addressed, traditional encryption methods and geographic location-based encryption methods. In each work relevant to what we are proposing, we analyzed the features and the shortcomings.

CHAPTER III

LOCATION AVAILABILITY

It is fair to say that most, if not all, VANET applications rely on accurate location information. The location information that is not available when you need it is almost as bad as none at all. Given the scale of its mobility and number of actors involved, the topology of a VANET is changing constantly and, as a result, both individual links and routing paths are inherently unstable. Therefore, in this chapter, we introduce a stable routing scheme to improve the location availability. Our main contribution is a probability model for link duration based on realistic vehicular dynamics and radio propagation assumptions. We then go on to show how the proposed model can be incorporated in a routing protocol that results in paths that are easier to construct and maintain, both locally and globally. Extensive simulation results confirm that our probabilistic routing protocol produces paths that are easier to maintain.

III.1 PROBABILITY ANALYTICAL MODEL

The main contribution of this section is to perform a probabilistic analysis of link duration in VANETs. Our results are based on a realistic distribution of headway distance as well as on a realistic channel model. Since, as already mentioned, there is no consensus in the literature on the exact distribution of the headway distance, we begin by an empirical validation of the log-normal headway distribution assumption proposed by Greenberg [85]. Our empirical validation serves the dual purpose of anchoring our analytical work in a realistic headway distance model and also to calibrate our subsequent simulation environment. Next, assuming a log-normal distribution of headway distance and a realistic channel propagation model, we derive the probability distribution as well as the expected duration of links in VANET. These analytical results are later confirmed by extensive simulation.

III.1.1 The Radio Model

The main goal of this section is to spell out the details of the radio propagation model that we assume throughout the thesis. It is well known that in wireless communications, signal attenuation is caused by the following three main factors [86]:

- *distance path loss* is the attenuation of the signal due to the distance between the transmitter and the receiver. Under most conditions, this is the principal cause of signal attenuation;

- *slow fading* (also known as *shadow fading* or *log-normal fading*) is caused by various factors including building dimensions, weather conditions and relative speed. As its name suggests, slow fading is typically modeled by log-normal distribution [87];

- *fast fading* (also referred to as *multi-path fading*) involves rapid fluctuations of the signal over small areas. Fast fading is mostly caused by the presence of reflectors, in the environment surrounding the transmitter or the receiver, which create multiple paths that the transmitted signal can traverse. As a result of multi-path, the receiver sees the superposition of multiple copies of the transmitted signal, each traversing a different path. Fast fading is typically modeled by a Rayleigh distribution [87].

Fast fading is short-term fading, and slow fading is long-term fading. Figure 12 shows the relationship among fast fading, slow fading and path loss. Slow fading captures long term fading effects and is generally applied in research and assumed as log-normal distribution [86]; fast fading is used in special scenarios where the coherence time of the channel is small relative to the delay constraint of the channel. Therefore, in the remainder of this chapter we are interested in path loss in the context of slow (i.e. shadow) fading only.

III.1.2 A Closer Look at Headway Distance in VANET

The *headway distance* between two vehicles is defined as the time (or, equivalently, distance) between two consecutive vehicles passing the same point and traveling in the same direction [88]. The headway distance between consecutive cars on a roadway plays a fundamental role, in understanding traffic flow, in ensuring travel safety and in other related issues. While there have been several attempts at suggesting a safe headway distance on roadways and streets, the task of legislating what the best headway distance should be is a complex problem fraught with technical, societal and political issues. This explains, to some extent, why the task of determining the probability distribution of headway distance is still an open question that has received a great deal of attention in the literature [85, 88, 89, 90, 91, 92, 93].

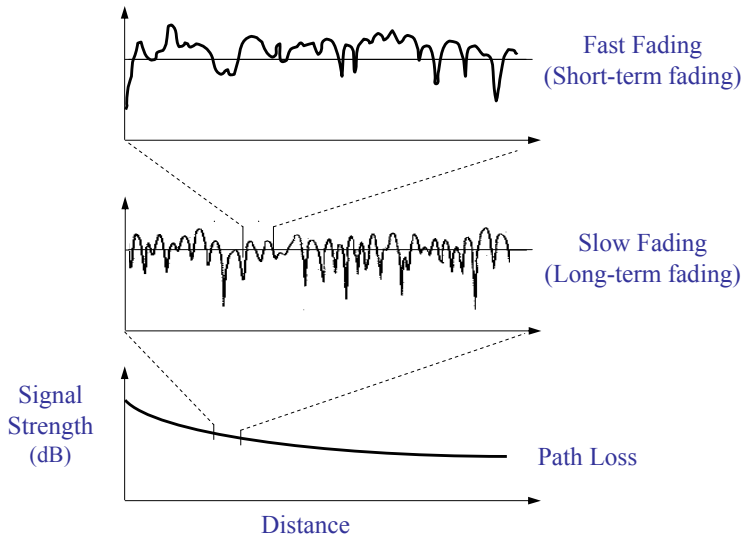


FIG. 12: Illustrating the relationship between slow fading, fast fading and path loss (with permission [4]).

Not surprisingly, many headway distance models have been developed since the 1960s. As pointed out by Cowan [91], typical representatives of such distribution models include the exponential distribution, the normal distribution, the gamma distribution and the log-normal distribution. For instance, the log-normal distribution was proposed to model headways under car-following situations [85]. A major assumption for the log-normal headway models is that a vehicle maintains a safe distance while following its leading vehicle closely at variable speeds. This assumption makes sense and is apparent in real traffic data [88, 89, 90, 94, 95, 96]. For example, Krbálek and Šeba [96] studied the statistics of public transportation in and around Mexico City. Chowdhury *et al.* [94] proposed a different distribution of headways. A road can be partitioned into a sections (boxes), that have the length of a car, i.e. 4 meters. The headways between two successive cars is defined based on the number of empty boxes between them. Abul-Magd [90] studied both the distribution of distance headway and time headway by applying the Beck and Cohen super-statistics [97]. Both traffic free phase and congested phase are studied. Some mixed distribution models are proposed on the assumption that a road consists of two components, tracking/following and free components. For example, Cowan [91] proposed a mixed distribution consisting of a constant distribution (tracking/following component) and

an exponential distribution (free component). Griffiths and Hunt [95] proposed another mixed model called the Double Displaced Negative Exponential Distribution (DDNED). These are common distributions of headway distance.

An Empirical Investigation of Headway Distance

Given the large variety of opinions in the literature concerning the probability distribution of the headway distance, we decided to begin our investigations by validating several candidates: the exponential distribution, the normal distribution, the gamma distribution and the log-normal distribution, in relation to their suitability as a basis for analytical studies of link distribution in VANET.

Towards this purpose, we have carried out experiments using the open source simulator written by Treiber [98]. Specifically, we have recorded and plotted the distance between consecutive cars. It is important to note that Treiber [98] does not have a specific model for the headway distance and, as a consequence, our empirical measurements were not biased towards any distribution. Having plotted the resulting headway distance, we then plotted, on the same graph, the various candidate probability distributions just mentioned (see Figure 13). For a detailed discussion we refer the reader to the Appendix.

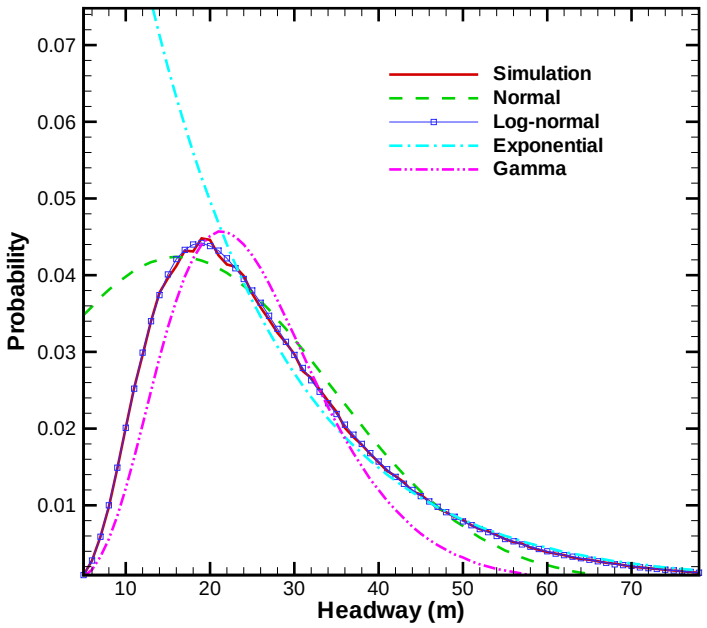


FIG. 13: The pdf of headway distance versus the normal, log-normal, exponential and gamma distribution. The log-normal distribution best matches our simulations.

As discussed below, and as revealed by Figure 14, we found that the best fit between a classic distribution function and the simulation results is provided by the *log-normal distribution*. Our results agree with those of a similar experiment conducted and reported independently by Puan [92].

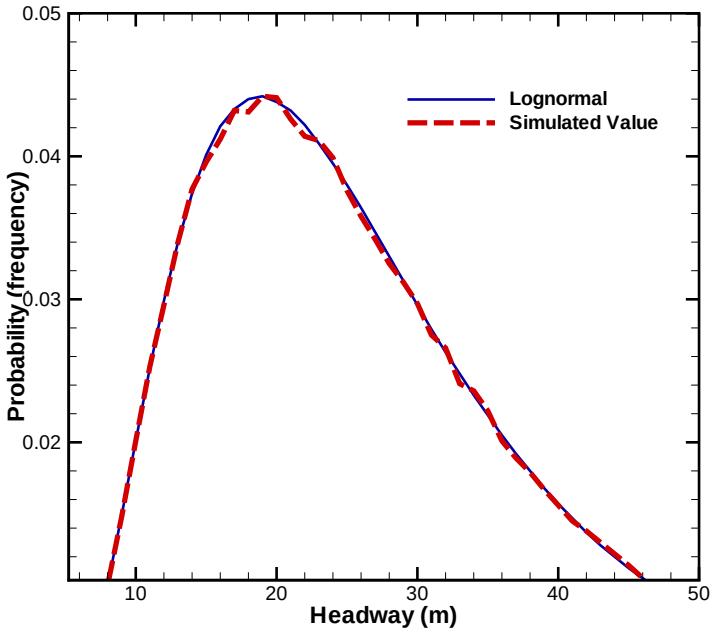


FIG. 14: Illustrating the pdf of headway distance versus the log-normal distribution

To fit the simulation curve of headway's probability density function (pdf) we solved a set of equations. The process of fitting the simulation curve of the headway's pdf is detailed in the Appendix. As is apparent from Figure 14, after curve fitting, the simulation results very closely match the log-normal distribution. The normal distribution is also similar to the simulation results. However, as evidenced in Figure 14, there are several segments where normal distribution does not match the simulation results, for example when headway distance is between 15-25m and also between 30-35m. We compared our simulation results with Paun's results. We found that both data are close to the log-normal distribution. As shown in Figure 15, our pdf is slightly shifted from Paun's data which was collected using video cameras to record traffic movement at four sites.

Given the good fit between our simulation results and the log-normal distribution, we have adopted the latter as the basis for our analytical derivations.

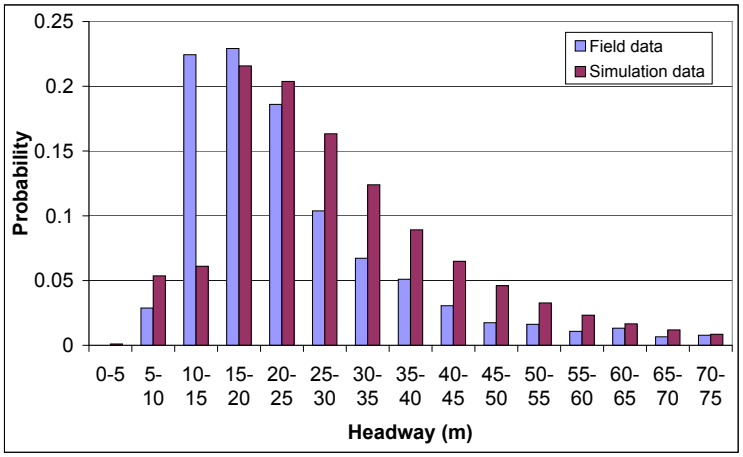


FIG. 15: Contrasting our headway simulation vs. Paun's field data.

The Log-normal Distribution – a Refresher

A continuous random variable X is said to have the log-normal distribution with parameters μ and σ if $X = e^Y$ where Y is normally distributed with parameters μ and σ. It is not hard to see that X itself is continuous and, for all positive x, the probability distribution function F_X of X reads

$$\begin{aligned} F_X(x) &= \Pr[\{X \leq x\}] \\ &= \int_0^x f_X(t)dt \\ &= \int_0^x \frac{1}{t\sigma\sqrt{2\pi}} e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}} dt \end{aligned} \tag{1}$$

where the probability density function (pdf) f_X is of the form

$$f_X(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}}. \tag{2}$$

For later reference, we recall that the expected value $E[X]$ and variance $Var[X]$ of the log-normal distribution with parameters μ and σ read

$$E[X] = e^{\mu+\frac{\sigma^2}{2}} \tag{3}$$

and

$$Var[X] = e^{2\mu+\sigma^2}\left(e^{\sigma^2} - 1\right). \tag{4}$$

III.1.3 The Probability Distribution of a Link

The Probability Distribution of Path Loss

The *path loss model* [86, 87, 99] is a radio propagation model that predicts the signal attenuation (in dB) at a distance X from the transmitter as a result of the combined effect of path loss and shadowing. More precisely, beyond a crossover distance d_0, the combined effect of path loss and shadowing is a random variable $L(X)$ described by

$$L(X) = l_0 + 10\gamma \log \frac{X}{d_0} + R_g \tag{5}$$

where[1]

- d_0 is the crossover distance, usually 1 meter [100];
- l_0 is the free-space path loss at distance d_0;
- $\gamma,\ (2 \leq \gamma \leq 6)$, is the power fall-off coefficient;
- the random variable R_g describes shadow fading; R_g is normally distributed with zero-mean and standard deviation σ_g, i.e. $R_g \in N(0, \sigma_g)$.

A quick look at (5) reveals that the path loss $L(X)$ is the convolution of two random variables: $Y = l_0 + 10\gamma \log \frac{X}{d_0}$ and R_g. It is important to note that, since we are interested in the stability of radio links, we assume that the distance X is a random variable that represents the headway distance between consecutive cars. We know X is log-normally distributed.

The probability density function of $L(X)$ obtained in the Appendix is $f(x)$,

$$f(x) = \frac{1}{(0.80x - 2.93)0.55\sqrt{2\pi}} e^{-\frac{(ln(0.80x-2.93)-2.95)^2}{2*0.55^2}}. \tag{6}$$

We are interested in the distance between a sender and a receiver. Let X_i be the i-th headway distance which is covered by the sender and the receiver, shown in Figure 16. Let there are m headway distances that are covered by the sender and

[1]Here, and in the remainder of this section, we use log to represent $\log_{10}$ and ln to represent the natural logarithm $\log_e$.

the receiver. $X_i \sim LogN(\mu_i, \sigma_i^2)$ be independent log-normally distributed variables with parameters σ_i and μ_i.

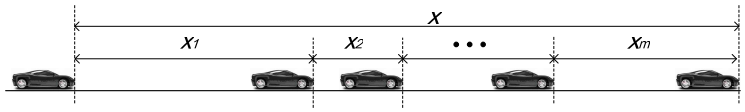


FIG. 16: Illustrating $X = X_1 + X_2 + \cdots + X_m$.

Let the link distance between the sender and the receiver be X_s. We notice $X_s = \sum_{i=1}^{m} X_i$. The distribution of X_s can be approximated by another log-normal distribution Z. The commonly-used Fenton-Wilkinson approximation [101, 102] is

$$\sigma_Z^2 = \log\left[\frac{\sum e^{2\mu_i+\sigma_i^2}(e^{\sigma_i^2}+1)}{(\sum e^{\mu_i+\sigma_i^2/2})^2} + 1\right]$$
$$\mu_Z = \log(\sum e^{\sigma_i^2}) - \frac{\sigma_Z^2}{2}.$$

To simplify the notation, we change variable Z to X with parameter μ and σ. Let $\mu = \mu_Z$ and $\sigma = \sigma_Z$. From this point, the variable X represents the sum of m headways where m is the number of headways included in the communication link. Therefore, the link distance variable is X. We assume that the safety distance between two vehicles is at least 6 meters and that the vehicles have an average length of 4 meters. The maximum number of headways in a wireless link with range 300 meters is, therefore, $\frac{300}{6+4}$, or 30, implying that $1 \leq m \leq 30$.

Lemma 1. *Assuming that X is log-normal with parameters μ and σ, the random variable $Y = l_0 + 10\gamma \log \frac{X}{d_0}$ is normally distributed.*

Proof. Let G_Y be the probability distribution function of Y. For every positive y, we write

$$G_Y(y) = \Pr[\{Y \leq y\}]. \tag{7}$$

Since Y is obviously continuous, we let g_Y stand for its density function. Now,

(7) allows us to write

$$\begin{aligned} G_Y(y) &= \Pr[\{Y \leq y\}] \\ &= \int_{-\infty}^{y} g_Y(t)dt \\ &= \Pr\{l_0 + 10\gamma \log \frac{X}{d_0} \leq y\} \\ &= \Pr\{X \leq d_0 10^{\frac{y-l_0}{10\gamma}}\} \\ &= \Pr\{X \leq 10e^{\frac{y-l_0}{10\gamma} + \log d_0}\} \\ &= \int_{-\infty}^{A(y)} f_X(t)dt \end{aligned} \tag{8}$$

where f_X is the density function of X and

$$A(y) = 10e^{\frac{y-l_0}{10\gamma} + \log d_0}.$$

We can obtain $g_Y(y)$ by differentiating (8) with respect to y.

$$\begin{aligned} g_Y(y) &= f_X\left(A(y)\right) \frac{dA(y)}{dy} \\ &= \frac{a}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(ay+b-\mu)^2}{2\sigma^2}\right) \\ &\in aN\left(\frac{\mu-b}{a}, \frac{\sigma^2}{a^2}\right) \end{aligned} \tag{9}$$

where $a = \frac{\ln 10}{10\gamma}$ and $b = \log d_0 \cdot \ln 10 - l_0 \frac{\ln 10}{10\gamma}$.

Thus, Y is normally distributed, completing the proof. □

Lemma 2. *Assuming that the headway distance is log-normally distributed, the path loss (including shadowing) has a normal distribution.*

Proof. Recall that, by (5), the path loss $L(X)$ is the convolution $L(X) = Y + R_g$. By Lemma 1, Y is normally distributed; as already mentioned, R_g is normally distributed with zero-mean and standard deviation σ_g, that is, $R_g \in N(0, \sigma_g)$. Moreover, Y and R_g are independent.

Now, it is a classic result in probability theory that the convolution of two independent normally distributed random variables is also normally distributed. The random variable $L(X) = Y + R$ is shown in Figure 17.

It follows that $L(X) = Y + R$ is normally distributed with pdf $l(z)$,

$$\begin{aligned} l(z) &\in aN\left(0 + \frac{\mu - b}{a}, \sigma_g^2 + \frac{\sigma^2}{a^2}\right) \\ &\in aN\left(\frac{\mu - b}{a}, \sigma_g^2 + \frac{\sigma^2}{a^2}\right) \end{aligned} \tag{10}$$

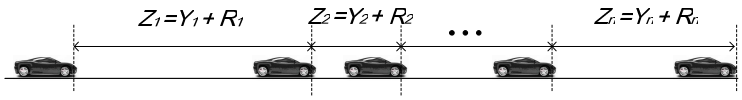


FIG. 17: Illustrating the random variables $L(X) = Y + R$

From (10) we can see that the mean path loss is $\frac{\mu-b}{a}$. To make the notation easier to read, we write $z = aN(\mu_1, \sigma_1^2)$, where $\mu_1 = \frac{\mu-b}{a}$ and $\sigma_1^2 = \sigma_g^2 + \frac{\sigma^2}{a^2}$. □

The Probability Distribution of the Existence of a Link

Our main goal in this section is to determine the probability distribution of a communication link established between two cars on a roadway. For this purpose, the basic approach we take is to evaluate the probability of a link based on path loss. Recall that we looked at the probability distribution of path loss under the shadow fading model. In this section, we shall often refer back to the notation established and to the results derived above.

The existence of a communication link between a transmitter vehicle and a receiving vehicle depends on the path loss at the receiver's side. For a link between the transmitter and the receiver to exist, the path loss between them needs to be smaller than a given threshold value PL_{thr}. Therefore, the probability distribution $F(z)$ of the existence of a link between a transmitter and receiver z distance units apart is:

$$\begin{aligned} F(z) &= P\{L(X) \leq z\} \\ &= \int_{-\infty}^{z} \frac{a}{\sigma_1\sqrt{2\pi}} \exp\left(-\frac{(t-\mu_1)^2}{2\sigma_1^2}\right) dt \\ &= \frac{C_1 a}{2}\left[1 + \text{erf}\left(\frac{z-\mu_1}{\sigma_1\sqrt{2}}\right)\right], \end{aligned} \tag{11}$$

where C_1 is a normalization coefficient and erf is the well know Gauss function (error

function) which is defined as

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt.$$

From the well-known relation

$$\lim_{z \to \infty} F(z) = 1,$$

we obtain $C_1 = \frac{1}{a}$. Consequently, we write

$$F(z) \quad = \tfrac{1}{2} + \tfrac{1}{2}\text{erf}\left(\tfrac{z-\mu_1}{\sigma_1\sqrt{2}}\right). \tag{12}$$

III.1.4 The Distribution Function of Link Duration

The duration of a link is the time interval during which an established communication link between two vehicles continues to exist. The main goal of this section is to derive analytical expressions for the (time) duration of a link and the probability distribution of the link duration.

Computing Link Duration

Assume, without loss of generality, that at time $t_0 = 0$, a communication link is established between two vehicles i and j moving in the same direction or in the opposite direction, with j ahead of i. Referring to Figures 18 and 19, let the random variable X denote the distance separating the two vehicles at link setup time. Mindful of the maximum DSRC transmission range constraint, we have

$$0 \leq X < 300. \tag{13}$$

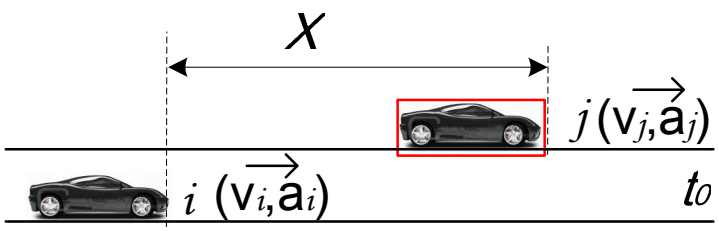


FIG. 18: Illustrating the same direction scenario.

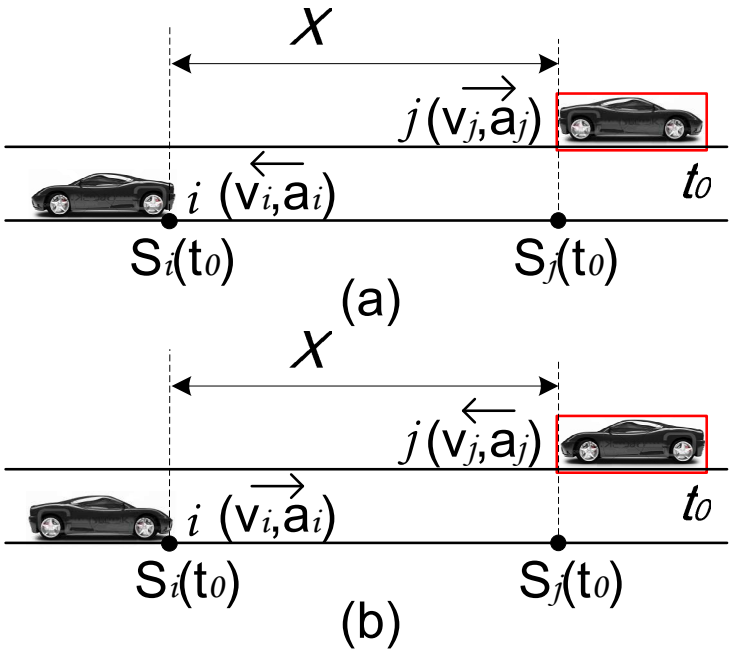


FIG. 19: Illustrating the opposite direction scenario.

It is important to note that X is the convolution of m independent headway distances with a common log-normal distribution. As discussed before, it is well known that the convolution of independent log-normal random variables can be approximated by a log-normal random variable [103, 104]. Thus, we assume that $X \in \text{logN}(\mu, \sigma)$ has parameters μ and σ.

We further assume that the speed limit on the roadway is v_m and that no vehicle will travel faster than v_m. For $t \geq 0$, we define $a(t)$, the acceleration of the vehicle at time t as follows:

- if $a(0) = 0$, then $a(t) = 0$ for all $t \geq 0$;
- if $a(0) > 0$, then

$$a(t) = \begin{cases} a(0) & \text{for } t \leq \frac{v_m - v(0)}{a(0)} \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

- if $a(0) < 0$, then

$$a(t) = \begin{cases} a(0) & \text{for } t \leq \frac{-v(0)}{a(0)} \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

In other words, (14) and (15) indicate that as long as the vehicle has not reached

the maximum speed v_m or has not stopped (in case $a(0) < 0$), its acceleration remains $a(0)$. However, once the vehicle reaches the speed limit (or has stopped), its acceleration becomes 0.

Given a generic vehicle with initial speed $v(0)$, the instantaneous speed $v(t)$ at time t is defined as

$$v(t) = v(0) + \int_0^t a(u)du, \tag{16}$$

where for all $u \in [0, t]$, $a(u)$ is the instantaneous acceleration at time u defined by above.

Now, (14) and (15) and (16) combined imply that

- if $a(0) = 0$, then $v(t) = v(0)$ for all $t \geq 0$;
- if $a(0) > 0$, then

$$v(t) = \begin{cases} v(0) + a(0)t & \text{for } t \leq \frac{v_m - v(0)}{a(0)} \\ v_m & \text{otherwise.} \end{cases} \tag{17}$$

- if $a(0) < 0$, then

$$v(t) = \begin{cases} v(0) + a(0)t & \text{for } t \leq \frac{-v(0)}{a(0)} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

Similarly, the distance that our generic vehicle travels in the time interval $[0, t]$ is defined as

$$S(t) = \int_0^t v(x)dx, \tag{19}$$

where $v(x)$ was defined above.

We now return to our vehicles i and j. To simplify the notation, we write $v_i = v_i(0)$, $a_i = a_i(0)$ and $v_j = v_j(0)$, $a_j = a_j(0)$. The instantaneous speeds and accelerations $v_i(t)$ and $a_i(t)$, respectively, $v_j(t)$ and $a_j(t)$ are obtained by suitably instantiating (14), (15), (17), and (18).

Now, (19) guarantees that the distances traversed in the time interval $[0, t]$ by vehicle i and j are, respectively,

$$S_i(t) = \int_0^t v_i(x)dx \tag{20}$$

and

$$S_j(t) = \int_0^t v_j(x)dx. \tag{21}$$

Assuming that at connection setup (i.e. time 0) the distance between the two vehicles was X, it follows that the distance between i and j at time t can be written as

$$S_j(t) - S_i(t) + X. \tag{22}$$

It is important to notice that (22) defines a *signed* distance: indeed, if at time t, $S_j(t) - S_i(t) + X > 0$, then vehicle j is ahead of i; otherwise, vehicle i is ahead of j.

For later reference, we find it convenient to define the indicator function $I(i, j)$ intended to capture information about which of the two vehicles is ahead when the communication link between them breaks:

$$I(i,j) = \begin{cases} 1 & \text{if } S_j(t) - S_i(t) + X > 0 \\ -1 & \text{otherwise.} \end{cases} \tag{23}$$

Given that DSRC links break at 300 meters, it follows that when the link breaks the following relation holds:

$$S_j(t) - S_i(t) + X = 300 \cdot I(i,j). \tag{24}$$

Figures 20, 21, and 22 which will be explained in detail below, illustrate the various possible combination of v_i, v_j, a_i, a_j. Figure 20 assumes that vehicles i and j move in the same direction, i.e. $v_i,\ v_j > 0$. Figure 21 shows the various combinations of v_i, v_j, a_i, a_j when i and j are moving in opposite direction, i.e. $\{v_i < 0, v_j > 0\}$ or $\{v_i > 0, v_j < 0\}$. Figure 22 illustrates that vehicles i and j decelerate to stop in either same direction or opposite direction. The two figures illustrate the relationship between speed (v-axis) and time (t-axis). The box beside the speed axis indicates a possible combination of v_i and v_j. There are three special times $t_\alpha, t_\beta, t_\gamma$ where t_α is the time when two vehicles have same speed, t_β is the time when one vehicle has zero speed but the other has nonzero speed and t_γ is the time when both vehicles stop. Due to the existence of a speed limit, we have t_ε and t_ζ as special time instants where t_ε is the time when one vehicle reaches the speed limit v_m and t_ζ is the time when both vehicles reach the speed limit. The reason that we discuss these special time moments is that these special time instants will change the definition of the

speed and velocity for vehicles and, thereafter, change the link duration time. The planar regions A', A'' and A''' are several sections of the area formed by speed lines. The planar regions will be used to derive the link duration as notations.

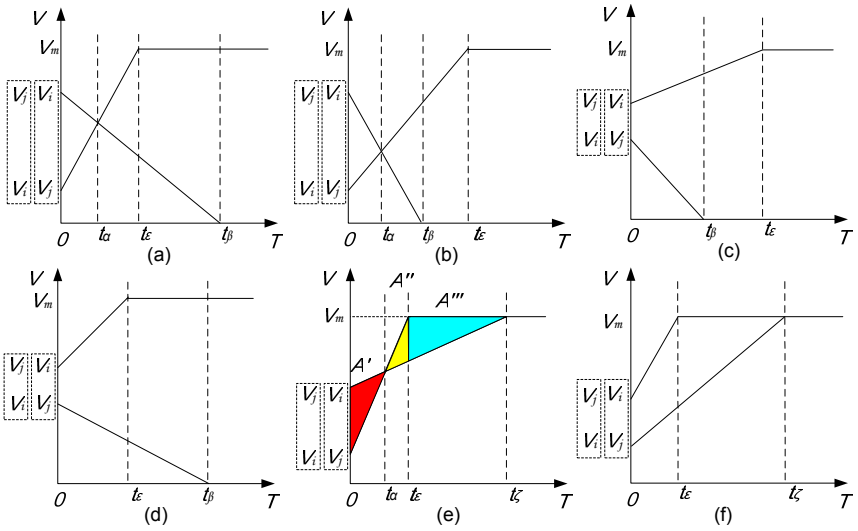


FIG. 20: Illustrating the same direction case.

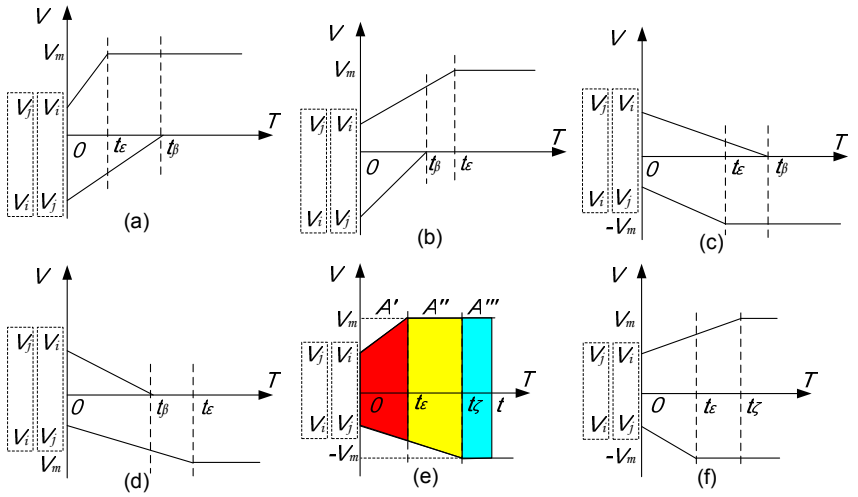


FIG. 21: Illustrating the opposite direction case.

We now define a number of time instances that will be used in our analysis:

- provided that $\frac{v_j - v_i}{a_j - a_i} > 0$, we write

$$t_\alpha = \frac{v_j - v_i}{a_j - a_i}; \tag{25}$$

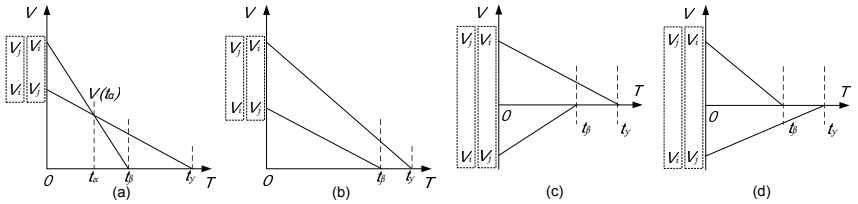


FIG. 22: Both vehicles decelerate and stop: a rare event in a highway scenario.

- define t_β as follows

$$t_\beta = \begin{cases} \frac{-v_i}{a_i} & \text{if } \frac{-v_i}{a_i} > 0 \text{ and } \frac{-v_j}{a_j} < 0 \\ \frac{-v_j}{a_j} & \text{if } \frac{-v_j}{a_j} > 0 \text{ and } \frac{-v_i}{a_i} < 0 \\ \min\{\frac{-v_i}{a_i}, \frac{-v_j}{a_j}\} & \text{if } \frac{-v_i}{a_i} > 0 \text{ and } \frac{-v_j}{a_j} > 0 \\ \text{undefined} & \text{otherwise;} \end{cases} \tag{26}$$

- similarly, define t_γ as follows

$$t_\gamma = \begin{cases} \max\{\frac{-v_i}{a_i}, \frac{-v_j}{a_j}\} & \text{if } \frac{-v_i}{a_i} > 0 \text{ and } \frac{-v_j}{a_j} > 0 \\ \text{undefined} & \text{otherwise;} \end{cases} \tag{27}$$

- define t_ε as follows

$$t_\varepsilon = \begin{cases} \frac{v_m - v_i}{a_i} & \text{if } \frac{-v_i}{a_i} > 0 \text{ and } \frac{v_m - v_j}{a_j} < 0 \\ \frac{v_m - v_j}{a_j} & \text{if } \frac{-v_j}{a_j} > 0 \text{ and } \frac{v_m - v_i}{a_i} < 0 \\ \min\{\frac{v_m - v_i}{a_i}, \frac{v_m - v_j}{a_j}\} & \text{if } \frac{v_m - v_i}{a_i} > 0 \text{ and } \frac{v_m - v_j}{a_j} > 0 \\ \text{undefined} & \text{otherwise;} \end{cases} \tag{28}$$

- define t_ζ as follows

$$t_\zeta = \begin{cases} \max\{\frac{v_m - v_i}{a_i}, \frac{v_m - v_j}{a_j}\} & \text{if } \frac{v_m - v_i}{a_i} > 0 \text{ and } \frac{v_m - v_j}{a_j} > 0 \\ \text{undefined} & \text{otherwise.} \end{cases} \tag{29}$$

It is important to note that $t_\alpha \leq t_\beta \leq t_\gamma$ and t_ε, t_ζ only depend on the speeds and accelerations of the two vehicles at connection setup time as well as on the value of the speed limit v_m. Therefore, vehicles i and j can use the following algorithm to

predict the time (if any) when the established link will break:

Algorithm III.1.1: FINDLINKBREAKTIME(v_i, v_j, a_i, a_j)

procedure PROBEROUTINGPATH$(time)$
 if $|S_j(time) - S_i(time) + X| \geq 300$
  then $t = +(0, time]$
  else $t = -(0, time]$
 return (t)

main
 $region = [0, +\infty]$
 $x \leftarrow$ use $v_i, v_j, a_i, a_j$to compute$t_\alpha, t_\beta, t_\gamma, t_\varepsilon, t_\zeta$
 for each $t \in x$
  do $region = region \cap$ BREAKTIMEREGION(t)
 determine the value of $I(x, y)$ based on the value of $region$
 solve the roots of the equation $S_j(t) - S_i(t) + X = I(i, j) \cdot 300$
 $time \leftarrow$ determine the right root
 output $(time)$

Discussion: The algorithm **FindLinkBreakTime** takes the four signed mobility parameters: v_i, v_j, a_i, a_j as input and returns the link duration time. First, $t_\alpha, t_\beta, t_\gamma, t_\varepsilon, t_\zeta$ are computed as described by (25) – (29) above. For each of them, we check the time region where the link may break. We mark the region where the link will break as a positive region and the region that the link will not break as a negative region. Then, we intersect these positive and negative regions to find the smallest region where the link will break. If it turns out that the connection breaks, the corresponding value of the indicator function $I(i, j)$ is computed. Finally, we can solve the equation to find the link duration time.

To make this work easier to read, we will discuss only two of the several possible cases, namely the one illustrated in Figure 20.e, as an illustrative example of the "same direction" scenario and the one illustrated in Figure 21.e as an example of the opposite direction scenario. Both sample cases are representative and prevalent on highways. We show how to compute the link duration given v_i, v_j, a_i, a_j as an illustration of the Algorithm III.1.1. A detailed derivation of the other cases can be found in [105]. The exact duration of the link in each of these cases is summarized

in the appendix ("Link Duration Cases").

Deriving the Link Duration When the Vehicles Move in the Same Direction

In this scenario we shall discuss only the case $v_j > v_i > 0$ in Figure 20.e as an example. In this example, both vehicles move in the same direction and have positive accelerations. Sooner or later they will reach the speed limit and will, thereafter, cruise at the maximum speed.

Case 1: $0 \leq t \leq t_\alpha$ The time region where the link breaks is A' in Figure 20.e. Since i, j are in A', when the link breaks vehicle j must be ahead of i and, thus, $I(i,j) = 1$. Recall that in DSRC the link breaks when the distance between i and j is 300 meters. Therefore, we can write

$$S_j(t) - S_i(t) + X = 300. \tag{30}$$

On the other hand, by (19)

$$\begin{aligned} S_j(t) - S_i(t) &= \int_0^t v_j(x)dx - \int_0^t v_i(x)dx \\ &= \frac{1}{2}a_j t^2 + v_j t - (\frac{1}{2}a_i t^2 + v_i t) \\ &= \frac{1}{2}a_r t^2 + v_r t. \end{aligned}$$

All that remains is to substitute $S_j(t) - S_i(t)$ in (30) and to solve for t. Since $t < t_\alpha$, we obtain

$$t = \frac{-v_r + \sqrt{v_r^2 + 2a_r(300 - X)}}{a_r}. \tag{31}$$

Case 2: $t_\alpha < t \leq t_\varepsilon$ In this case, the time region where the link breaks is A'' in Figure 20.e. Since i, j are in A'', when the link breaks vehicle i must be ahead of vehicle j; thus $I(i,j) = -1$ and by (24) we can write

$$S_j(t) - S_i(t) + X = -300. \tag{32}$$

We know that $S_j(t) - S_i(t) = \frac{1}{2}a_r t^2 + v_r t$. By substituting the value of $S_j(t) - S_i(t)$ in (32) and by solving for t we obtain

$$t = \frac{-v_r - \sqrt{v_r^2 - 2a_r(300 + X)}}{a_r}. \tag{33}$$

Case 3: $t_\varepsilon < t \leq t_\zeta$ In this case, the time region where the link breaks is denoted by A''' in Figure 20.e. Since i, j are in A''', when the link breaks vehicle i must be ahead of vehicle j and so $I(i, j) = -1$. Thus, we can write

$$S_j(t) - S_i(t) + X = -300 \tag{34}$$

Observe that by (19), $S_j(t) = \frac{1}{2}a_j t^2 + v_j t$ and $S_i(t) = v_m t - \frac{v_m - v_i}{2}t_\varepsilon$. After substituting the value of $S_j(t) - S_i(t)$ in (34) and after solving for t we obtain

$$t = \frac{-(v_j - v_m) - \sqrt{(v_j - v_m)^2 - 2a_j(300 + x + \frac{v_m - v_i}{2}t_\varepsilon)}}{a_j}. \tag{35}$$

Case 4: $t_\zeta < t$ In this case, the link will not break because the two vehicles are moving in the same direction and at the same speed. Therefore, in this case, the connection breaks at $+\infty$. In reality, the $+\infty$ time means an extremely longer time compared with the average link duration.

The situation in Figure 20.e shows that vehicle i catches up vehicle j, passes j and finally breaks the link with vehicle j. This is a quite frequent case on the highway.

Deriving the Link Duration When the Vehicles Move in Opposite Directions

In this scenario we shall discuss only the case $v_j > v_i > 0$ in Figure 21.e where vehicles i and j are both accelerating and reaching speed limit while moving in opposite directions.

Case 1: $0 < t \leq t_\varepsilon$ The time region where the link breaks is denote by A' in Figure 21.e. Since i, j are in A', when the link breaks vehicle j must be ahead of vehicle i and, consequently, $I(i, j) = 1$. We can write

$$S_j(t) - S_i(t) + X = 300. \tag{36}$$

It is easy to see that $S_j(t) - S_i(t) = \frac{1}{2}(a_j - a_i)t^2 + (v_j - v_i)t = \frac{1}{2}a_r t^2 + v_r t$. After substituting this value in (36), we can solve for t to get

$$t = \frac{-v_r + \sqrt{v_r^2 + 2a_r(300 - X)}}{a_r}. \tag{37}$$

Case 2: $t_\varepsilon < t \leq t_\zeta$ In this case, the time region where the link breaks is denoted by A'' in Figure 21.e. Since i, j are in A'', the vehicle j must be ahead of vehicle i when the link breaks, thus $I(i, j) = 1$. We must have

$$S_j(t) - S_i(t) + X = 300. \tag{38}$$

It is easy to confirm that, in this case, $S_j(t) = v_m t - \frac{v_m - v_j}{2}t_\varepsilon$ and $S_i(t) = \frac{1}{2}a_i t^2 + v_i t$ since j has already reached the speed limit v_m ($v_j = v_m$ and $a_j = 0$). By substituting these expressions in (38) and by solving for t we get

$$t = \frac{-(v_i - v_m) - \sqrt{(v_i - v_m)^2 - 2a_i(300 - X + \frac{v_m + v_j}{2}t_\varepsilon)}}{a_i}.$$

Case 3: $t_\zeta < t$ In this case the time region where the link breaks is denoted by A''' in Figure 21.e. Since i, j are in A''', the vehicle j must be ahead of vehicle i when the link breaks, and so $I(i, j) = 1$. When the link breaks, we can write

$$S_j(t) - S_i(t) + X = 300 \tag{39}$$

Observe that in this case $S_j(t) = v_m t - \frac{v_m - v_j}{2}t_\varepsilon$ and $S_i(t) = -v_m t - \frac{-v_m - v_i}{2}t_\zeta$ because both vehicles i and j have reached the speed limit. After substituting $S_j(t)$ and $S_i(t)$ and after solving (39) we obtain

$$t = \frac{300 - X + t_\varepsilon(v_m - v_j)/2 + t_\zeta(v_m + v_i)}{2v_m}. \tag{40}$$

This case shows that vehicle j and i move in opposite directions and are accelerating to reach the maximum speed. This is a quite frequent case on the highway.

Reasoning About Link Duration Distribution

With the preamble of the previous section out of the way, we are now ready to state and prove the following important result.

Lemma 3. *Assuming that X is log-normal with parameters μ and σ, a random variable $T = \sqrt{aX + b} + c$ is log-normally distributed, where $a, b, c \in \mathbb{R}$, $a, b, c \neq 0$ and $aX + b \geq 0$.*

Proof. Let G_T be the probability distribution function of T. For every positive t, we write

$$G_T(t) = \Pr[\{T \leq t\}]. \tag{41}$$

Since T is obviously continuous, now, (41) allows us to write

$$\begin{aligned}
G_T(t) &= \Pr[\{T \leq t\}] \\
&= \Pr[\{\sqrt{aX + b} + c \leq t\}] \\
&= \Pr[\{aX \leq (t - c)^2 - b\}] \\
&= \begin{cases} \Pr[\{X \leq \frac{(t-c)^2 - b}{a}\}] & \text{for } a > 0 \\ \Pr[\{X \geq \frac{(t-c)^2 - b}{a}\}] & \text{for } a < 0 \end{cases} \\
&= \begin{cases} F_X\left(\frac{(t-c)^2 - b}{a}\right) & \text{for } a > 0 \\ 1 - F_X\left(\frac{(t-c)^2 - b}{a}\right) & \text{for } a < 0 \end{cases}
\end{aligned}$$

where F_X is the probability distribution function of X. When $a > 0$, it is clear that T is log-normally distributed.

Next, we propose to show that T is also log-normally distributed when $a < 0$.

For this purpose, using (12), we write

$$\begin{aligned}
1 - F_X\left(\frac{(t-c)^2-b}{a}\right) &= 1 - \frac{1}{2} - \frac{1}{2}erf\left(\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right) \\
&\quad [\text{where } z = \tfrac{(t-c)^2-b}{a}] \\
&= \frac{1}{2}\left[1 - erf\left(\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(-\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&\quad [\text{since } -erf(x) = erf(-x)] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{-\ln z + \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{\ln \frac{1}{z} + \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{\ln \frac{a}{(t-c)^2-b} - (-\mu(X))}{\sigma(X)\sqrt{2}}\right)\right] \\
&= F_Y\left(\frac{a}{(t-c)^2-b}\right), \qquad (42)
\end{aligned}$$

where Y is a log-normal random variable with parameters $-\mu(X)$ and $\sigma(X)$. Thus, in all cases, T obeys a log-normal distribution, completing the proof. □

Lemma 4. *Assuming that X is log-normal with parameters μ and σ, the random variable $T = aX + b$ is log-normally distributed, where $a, b, c \in \mathbb{R}$ and $a, b, c \neq 0$.*

Proof. Let G_T be the probability distribution function of T. For every positive t, we write

$$G_T(t) = \Pr[\{T \leq t\}]. \qquad (43)$$

Now, (43) allows us to write

$$\begin{aligned}
G_T(t) &= \Pr[\{T \leq t\}] \\
&= \Pr[\{aX + b \leq t\}] \\
&= \Pr[\{aX \leq t - b\}] \\
&= \begin{cases} \Pr[\{X \leq \frac{t-b}{a}\}] & \text{for } a > 0 \\ \Pr[\{X \geq \frac{t-b}{a}\}] & \text{for } a < 0 \end{cases} \\
&= \begin{cases} F_X\left(\frac{t-b}{a}\right) & \text{for } a > 0 \\ 1 - F_X\left(\frac{t-b}{a}\right) & \text{for } a < 0 \end{cases}
\end{aligned}$$

where F_X is the probability function of X. When $a > 0$, we observe that T is log-normally distributed. It remains to be shown that T is log-normally distributed even if $a < 0$. From (12), we write

$$\begin{aligned}
1 - F_X\left(\frac{t-b}{a}\right) &= 1 - \frac{1}{2} - \frac{1}{2}erf\left(\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right) \\
&\quad [\text{where } z = \tfrac{t-b}{a}] \\
&= \frac{1}{2}\left[1 - erf\left(\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(-\frac{\ln z - \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&\quad [\text{since } -erf(x) = erf(-x)] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{-\ln z + \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{\ln \frac{1}{z} + \mu(X)}{\sigma(X)\sqrt{2}}\right)\right] \\
&= \frac{1}{2}\left[1 + erf\left(\frac{\ln \frac{a}{t-b} - (-\mu(X))}{\sigma(X)\sqrt{2}}\right)\right] \\
&= F_Y\left(\frac{a}{t-b}\right) \qquad (44)
\end{aligned}$$

where Y is a log-normal random variable with parameters $-\mu(X)$ and $\sigma(X)$. Thus, in all cases T is log-normally distributed, completing the proof. □

Lemma 5. *Suppose that the communication link between two vehicles i and j breaks at time t. The link duration time is either a linear function of X or a square root function of X.*

Proof. Recall that when the link breaks, t satisfies (24). By the definition of $S_i(t)$, we know that $S_i(t) = \int_0^t v_i(x)dx$ is a linear function of t when the speed v_i is constant, i.e. $v_i(t) = v_m$. Let $S_i(t) = at + b$. Similarly, $S_j(t)$ is a linear function of t when v_j is constant, i.e. $v_j(t) = v_m$. Let $S_j(t) = ct + d$. When both $S_i(t)$ and $S_j(t)$ are linear function of t, substituting the corresponding values of $S_j(t)$ and $S_i(t)$ in (24), we obtain

$$\begin{aligned}
(a-c)t + b - d + X &= I(i,j)300 \\
\Rightarrow t &= \frac{I(i,j)300 - X - b + d}{a - c}. \qquad (45)
\end{aligned}$$

The t can be a linear function when both $v_i(t)$ and $v_j(t)$ are constant. If any of $v_i(t)$ and $v_j(t)$ is not a constant function, the distance function will be a polynomial function with degree 2. Without loss of generality, we let $v_i(t) = v_i(0) + a_i t$, by definition, the distance function

$$\begin{aligned} S_i(t) &= \int_0^t v_i(0) + a_i x dx \\ &= v_i(0)t + \frac{1}{2}a_i t^2. \end{aligned} \tag{46}$$

Therefore, $S_j(t) - S_i(t)$ will be a quadratic polynomial. Suppose $S_j(t) - S_i(t) = at^2 + bt + c$ and $a \neq 0$. Substitute $S_j(t) - S_i(t)$ in (24) to get the quadratic equation

$$at^2 + bt + c + X - I(i,j)300 = 0. \tag{47}$$

Clearly, t exists because the link breaks at time t. Therefore, the solution must be a square root function of X. This completes the proof. □

Theorem 1. *The duration of the link between vehicles i and j is log-normally distributed.*

Proof. By Lemma 5, the link duration can be expressed as one of the two formulas: $aX + b$ or $\sqrt{aX + b} + c$. By Lemma 3, the expression $\sqrt{aX + b} + c$ is log-normally distributed. By Lemma 4, $aX + b$ is log-normally distributed. Thus, in all cases, the duration of the link has a log-normal distribution. This completes the proof of the theorem. □

The Mean of Link Duration

Let Φ be a set of all real combinations of v_i, v_j, a_i, a_j on roads and ϕ be the size of Φ. Let P_k be the probability of the case $k \in \Phi$ and T_k be the link duration of case k. By the law of total expectation, we can obtain the overall expected duration of a link $E(link)$,

$$E(link) = \sum_{k=1}^{\phi} P_k T_k. \tag{48}$$

Theoretically, the $E(link)$ can be computed by (48). But there are no analytical results or field-tested data on P_k and T_k in the literature. We will leave the computation of the expected link duration as future work. Once we have the probability function of a link, we can easily compute the probability of a link by suitably instantiating t in $G_T(t)$. Specifically, the t is usually the routing path duration requirement $EDur$.

III.2 OUR PROPOSED PROTOCOL

The main contribution of this section is to show how the proposed routing protocol works on the basis of the above probability model. We assume that the system is composed by vehicles which are installed with transceivers, GPS and unique identities. For the purpose of this work, we assume that the traffic is moderate in intensity, since sparse traffic renders networking as we describe it impossible. We also assumed that only the vehicles on the same direction will be recruited to propagate packets because the oncoming vehicles break the routing link sooner than the same directional vehicles.

III.2.1 Probing the Routing Path

Due to high mobility, the topology of the network is constantly changing. Therefore, a reactive routing path probing is needed, i.e. routing path searching is on-demand, for example, DSDV. In the literature, there are several reactive routing protocols, for example AODV and DSR etc. which were discussed in Chapter II.1.2. These protocols find a routing path on the basis of reachability instead of quality of the routing path. We propose a new routing path search protocol on the basis of quality of the routing path: the duration of the routing path, as well as the reachability. This is because the duration of a routing path is the most difficult and important metric to consider when routing packets in the presence of high vehicular mobility.

In the routing process illustrated in Figure 23.a, we use the unique vehicle ID as the network address. The source vehicle will send out a probing request, a control packet: Prb. Since the Prb is broadcast by the wireless channel, all vehicles in the transmission range will receive it and compute the distance from the sender, the duration of the link and the probability of link duration. The computation is based on the previous discussion. An acknowledgment packet $AckP$ will be constructed by

using the computed distance, duration, and probability of link duration. The *AckP* is sent back to the sender who will collect multiple *AckP*s and select the best link. The best link is the one with highest probability and larger duration than the routing path duration requirement *EDur*. The sender only sends a confirmation packet *CfmP* to the best link, which will further explore the routing path by multicasting the *Prb*. When the destination node receives the probing packet, it will terminate the probing and send an acknowledgment packet back to the source node along the newly formed routing path. This completes the routing path exploration stage. The route we found is optimal in terms of duration length.

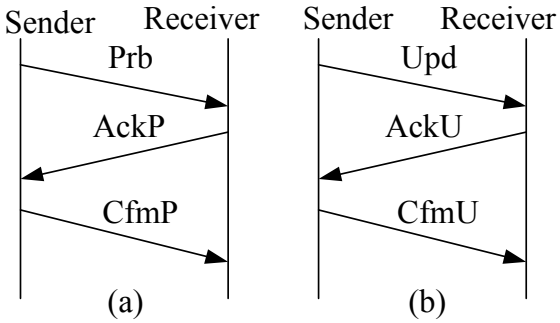


FIG. 23: The proposed protocol

type	dur	EDur	Pr	hop
Src	Dst	ID	pre	next
r_v	r_a	s_v	s_a	dis
Others (len, chksum, priority, sig, etc)				

FIG. 24: The packet header

III.2.2 A Summary

The following pseudo-code is intended to summarize our previous discussion:

Algorithm III.2.1: RoutingPathProtocol(*packet*)

```
main
 while p ← receive a packet
       do { if p.type = Prb or p.type = Upd
              then { t ← FindLinkBreakTime(v_i, v_j, a_i, a_j)
                     Pr ← compute the probability of link duration
                     d ← compute the distance from the sender
                     construct a ACK ← t, d, Pr
                     send the ACK to the sender respectively
            if p.type = AckU
              then { select the hop with highestPr and t < EDur
                     send a confirm packet CfmU
            if p.type = AckP
              then do { select the hop with highestPr and t < EDur
                        send a confirm packet CfmP
            if p.type = CfmP
              then do { if p.hops < MaxHop
                          then
                        broadcast a Prb message to find the next hop
                          else send ERR message to sender
            if p.type = CfmU
              then do { if p.hops < MaxHop
                          then
                        broadcast a Upd message to find the next hop
                          else send ERR message to sender
            drop message p
```

Discussion: The algorithm **RoutingPathProtocol** involves an infinite loop. In each iteration, it takes one packet, parses the type of the packet and processes the

packet according to its type. Packets of type Prb or Upd are processed similarly: 1) computing the link duration, the probability of a link duration with the $EDur$, and the distance from the sender; 2) sending an acknowledgment packet $AckP$ if the packet is Prb or $AckU$ if the packet is Upd. If the type of the packet is $AckP$ or $AckU$, we will select the hop that satisfies the condition that the link duration is greater than the path duration requirement, and that is a hop with highest probability computed by method in Section III.1.3 using $EDur$. If the type of the packet is $CfmP$ or $CfmU$, the protocol will continue to find the next hop. From the above discussion, we can observe that the probing of the routing path is processed by the control packet Prb. The maintenance of the routing path is done by the control packet Upd.

III.2.3 Routing Path Maintenance

In this section, we are going to address two routing path maintenance methods, local repair and global repair. First, the links that cause the breakage of the routing path will be locally repaired/replaced. Second, a backup route will be constructed globally before the breakage of the current routing path and will be switched as the current route. The repair procedures are based on the prediction of the link duration and the probability of the link duration discussed in Section III.1.3. By using this prediction, we can repair the current path before it is broken. In this way, we enhance the reliability of the routing path and provide location availability.

Local Repair

In our protocol, data packets will include a mobility header which includes: the type of the packet ("type" in Figure 24), the duration of the link ("dur"), the routing path duration requirement ("EDur"), the probability of the link ("Pr"), the number of hops ("Hop"), the source and destination address ("Src", "Dst"), the current node's id ("ID"), the previous and next router ("pre","next"), the receiver of the link's signed speed and acceleration ("r_v","r_a"), the sender of the link's signed speed and acceleration ("s_v","s_a"), the distance of the link ("dis"). During the communication, each receiver of a link will monitor the connectivity condition of the link by computing the duration and the probability of the link. If the link is about to break, the receiver will send a control packet $RBRK$ (Routing BReaK). The sender will send a Upd packet whose hop is set to 1. The Upd packet only explore the one-hop neighbors to find a replacement link.

Global Repair

All the vehicles along the routing path know the expected expiration time of the path. These vehicles do not need to check the connectivity of the routing path if the vehicles move at constant speed. But the prediction of the routing path needs to be updated due to the dynamics of vehicular mobility where vehicles accelerate and decelerate randomly. Therefore, the connectivity of the routing path needs to be updated as well.

The frequency of the updating packets t_{upd} is determined as follows. There is a fixed frequency, t_{uf} for example, 10 seconds. Consider that there will be some short connection. Suppose p_{pnt} is a certain percentage of the routing path duration requirement *EDur*. The frequency of updating packets is t_{uf} if $t_{uf} \leq p_{pnt} * EDur$, otherwise, it is $p_{pnt} * EDur$ if $t_{uf} > p_{pnt} * EDur$.

As shown in Figure 23.b, the source node sends a update packet Upd every t_{upd} seconds. When a node receives the Upd, it computes the distance from the sender, the duration of the link and the probability of the link. The computation is based on the previous discussion and will not be repeated here. An acknowledgment packet $AckU$ is constructed using the computed distance, duration, and probability of the link. The $AckU$ is sent back to the sender who will collect multiple $AckU$s and will select the best link which is the one with highest probability and having larger duration than the routing path duration requirement *EDur*. The sender only sends a confirmation packet $CfmU$ to the best link, which will further explore the routing path by multicasting the Upd. When the destination node receives the update packet, it will terminate the updating process and will send an acknowledgment packet back to the source node along the latest routing path. This completes the routing path update.

III.3 SUMMARY

In this chapter, we addressed a fundamental problem, namely that of enhancing location availability. Due to the high vehicle mobility, the topology of VANET changes constantly. The changing topology makes the wireless connection (routing) difficult and makes location availability extremely challenging. Our motivation is to ensure location availability over a certain period of time by computing the duration of a routing path and the probability of the path. To compute the two parameters for

the routing path, we compute the two parameters for the routing links which compose the routing path.

We first discussed the probability analytical model which is the basis of our wireless routing protocol. The motivation of the model is to derive the expression of link duration and the distribution of link probability. We started from the concept of headway distance. Given the mobility of each vehicle, we can compute the link duration. With the fact that headway distance is log-normally distributed, we can show that the distribution of link duration is log-normally distributed as well. Once we have the two expressions of link duration and link probability, we can construct a routing path. Since the duration of our routing path is predictable, we can reduce the maintenance control messages and achieve good status of the routing path. Therefore, we can reduce the control message overhead and improve the response time of messages (by avoiding rebuilding a routing path).

CHAPTER IV

LOCATION INTEGRITY

We present validation mechanisms to provide location *integrity* in VANETs. In our approach, we use network cells as both security and communication units. Providing location integrity is thus split into intra-cell integrity and inter-cell integrity. We provide three solutions to intra-cell integrity; each of the three solutions assumes different on-board equipped devices. First, we assume that all vehicles are equipped with an on-board radar device, a GPS unit and a standard transceiver. To ensure intra-cell location information integrity, we propose an active validation solution (called *active location integrity*) which relies on the help of on-board radar to detect neighboring vehicles and to confirm their alleged coordinates. Since radar is not currently installed in all vehicles, we propose a second solution (called *passive location integrity*) which relies on information fusion to filter out malicious data and refine low-resolution location information into high-resolution location information. Mindful of the fact that some of the vehicles participating in the traffic may not have any of these devices, we propose a third validation solution (called *general location integrity*) which combines the active and passive location integrity solutions. Since VANET applications often need the location information of vehicles that belong to different cells, we address inter-cell location information integrity as well. Vehicles request that their neighbors or vehicles in oncoming traffic check the alleged location information of remote vehicles. Both the request and response messages are propagated among the cells.

The validated location obtained from the above mechanisms is stored in the memory of vehicles in a certain time interval. The stored locations constitute a location history, called *Map History*. The map history can further improve the location integrity. The map history can be used to distinguish a "real" location from a "fabricated" location. The basic idea is that any vehicle without historical consistency is highly suspect. All location information must be consistent with the location history. The vehicles which send fabricated locations will be isolated by using a trust mechanism. If a vehicle is isolated, the vehicle will not be able to receive or send messages from or to other vehicles. We will also address the trust mechanism in this chapter.

IV.1 CELL-BASED NETWORK

Based on the philosophy of "divide and conquer", a cell-based network is adopted in this dissertation. The advantage of this network is local optimization for both location integrity and location availability. For integrity, we can use an "eye device" (i.e. radar) to validate the location and achieve local optimization of location integrity inside a single cell. Once we achieve local optimization, we can extend the local optimization to global, i.e. inter-cell optimization. In this chapter, we assume a medium level of traffic density. Scenarios of sparse traffic or traffic with gaps are our future work.

IV.1.1 Network Formation

Network communication can be modeled either as a city scenario or a highway scenario. We choose to address the highway scenario because we are concerned with security, and Leinmüller *et al.* [106, 107] showed that the effects of malicious nodes in a highway scenario are worse than in a city scenario.

Network Cells

Two types of cells have been proposed in the literature [26, 75]: dynamic cells and location-based cells. Dynamic cells are vehicle clusters that are formed on the fly based on the current location of vehicles. Location-based cells are specific regions that are pre-defined on the digital map. Although dynamic cells are flexible, they are not efficient. Location-based cells, on the other hand, are created in advance, and vehicles use their GPS coordinates to map themselves to their respective cells. These preset cells avoid the need to undergo the complex process of forming a cell. In this dissertation, we use location-based cells to build a communication network. An example of two cells is shown in Figure 78. The center of a cell is in the median between the two directions of traffic. We assume the two directions of traffic in the same cell can always communicate with each other, i.e. there are no large obstacles between the two directions.

The Advantage of Cells

In VANETs, routing protocols are often based on flooding because the topology changes very frequently. In flooding, every incoming packet is sent out by wireless

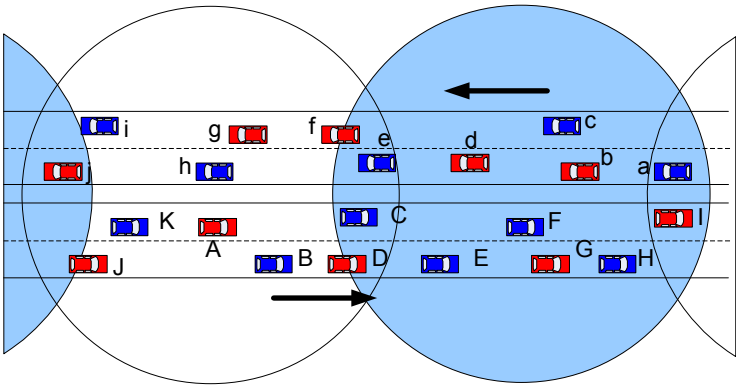


FIG. 25: System model. The circular areas are the preset cells. A vehicle compares its GPS coordinates with these preset cells to identify its host cell.

broadcasting. No routing table are necessary. The major problem with flooding is the vast number of duplicates packets. Some means of controlling the unchecked growth of the number of packets is needed.

The reason we use cells is to save bandwidth, reduce delay and increase security. We try to avoid flooding messages over the whole network all the time. Bandwidth is wasted by flooding messages because all nodes are involved in relaying and communicating. Significant delays can be caused by the collision of flooding messages and even the broadcast storm [108] may arise in dense traffic. Last but not least, the more vehicles that are involved in forwarding the message, the more likely the message is modified.

We want to reduce the number of routing participants. Therefore, we partition the network into small cells. In each cell, we locally manage the information. Only a cell leader and a cell router respond to the request from inter-cell nodes. The rest of the nodes in the cell keep silent but monitor the behavior of the cell leader and cell router. The cell-based structure improves the scalability of the system. In flooding, the number of duplicates will exponentially increase with the distance of message propagation. In cell-based routing, for each hop, there are only one or two cell routers involved. It dramatically reduces the number of duplicate messages.

Formation of Network Cells

We configure the road with virtual digital cells. For example, every 300 meters there is a cell on the road, i.e., cell's radius is 150 meters. All vehicles inside a cell are one-hop neighbors. The diameter of cells matches the transmission range of radar, so that all the neighbors inside can be directly detected by radar when the line of sight is clear. In this dissertation, we use a highway model and assume that cell overlap is about 30 meters. When vehicles are close to the overlap area between two cells, they may be chosen as routing vehicles as will be discussed in Section IV.1.2.

The series of cell center coordinates are $(Xc_i, Yc_i), 0 < c_i < n$. A vehicle is at coordinate (x_v, y_v) on the highway. The computer in the vehicle will find the closest cell center coordinates (Xc_k, Yc_k) to the vehicle's coordinates. If the coordinates satisfy $(x_v - Xc_k)^2 + (y_v - Yc_k)^2 \leq 150^2$, the vehicle is in cell k.

IV.1.2 Message Propagation

Location messages can be propagated over the cell-based network. For example, a observer vehicle A wants to know the topology of a certain cell whose cell leader is B. To save bandwidth, the request message is not propagated by all vehicles but only by cell leaders and cell routers. The request will be delivered to B. The cell leader B will send back topology packets which can be used by the observer to build a rough topology of the network cells. The message routing process is shown in Figure 26.

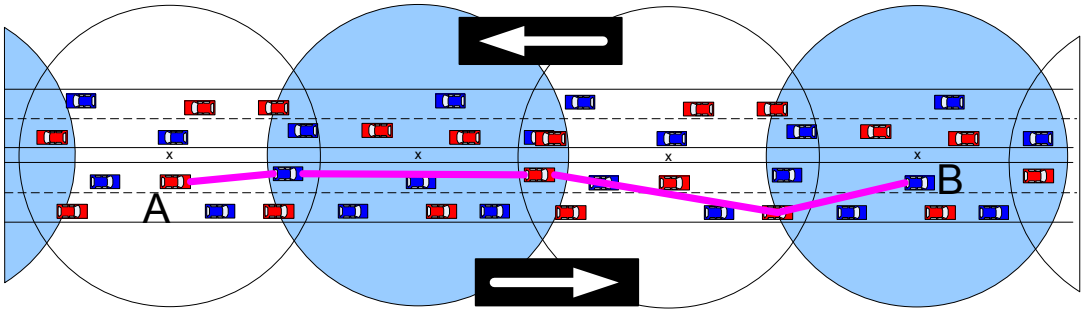


FIG. 26: Message propagation over the cell based network.

Cell Leader

The main duties of the cell leader are to verify the GPS location of all the vehicles in its cell, aggregate these locations and broadcast this data to other vehicles in the

cell. The cell leader collects the locations of all vehicles in its cell and exchanges this location information with other cells.

If a leader is determined to be malicious, the leader can be challenged by another member of the cell. A vehicle challenging to become the new leader must meet one of the following conditions: 1) be closest to cell center, 2) be approaching the cell leader. Condition 1 has precedence over condition 2. We can use the following formula to compute the score s for each leader candidate

$$s = C_{dc} * D_c + C_{di} * D_i \tag{49}$$

where C_{dc} is a coefficient for the distance between the vehicle and the center of the cell, D_c is the distance to the cell center, C_{di} is a coefficient for the traveling direction, and D_i is the traveling direction. If the vehicle is moving towards the center of the cell then $D_i = 1$. If it is moving away from the center then $D_i = -1$. We define $C_{dc} = \frac{R}{D_c+1}$ and $C_{di} = R$. The candidate with highest score wins.

If a leader is about to exit the cell, the leader will send an announcement to inform other vehicles. The other vehicles will elect a new leader to replace the existing one. The algorithm of electing a new leader is same as the challenging algorithm stated above. Therefore, a leader normally can stay in action for about 150m. The advantage of this cell-based network is lower risk of attacks. Even if there are attackers who temporally are cell leaders, they can only harm the system during a short period of time. The drawback of this cell-based network is control message overhead. But the control messages are local, i.e. inside a cell. There is not much effect on inter-cell communication.

Cell Router

The purpose of a cell router is to propagate messages from cell to cell and to reduce the number of duplicate messages. Determination of cell routers is similar to the determination of a cell leader. The difference is that two cell routers (upstream and downstream) will be determined instead of only one. Routers should be far from the center of the cell and close to the overlapping region. Cell routers can also be selected on the basis of link duration and link probability which are computed by a probability model discussed in Chapter III. The basic idea is as follows: the cell leader wants to send an aggregated location message to other vehicles which are

outside of the cell leader's cell. The cell leader will compute the link duration and link probability among all neighboring vehicles. Then, the cell leader will select a neighboring vehicle which has the largest link duration and link probability value, as the cell router.

IV.2 ACTIVE LOCATION INTEGRITY

In this section, we assume that all vehicles have on-board radar, GPS, and transceiver. The network is a homogenous system, i.e. all vehicles are installed with the same devices. In this scenario, we will address our first solution active location integrity. Underlying our solution is the famous adage: "*Seeing is believing*". We use on-board radar as the virtual "eye" of a vehicle and a wireless transceiver as the virtual "ear". Although the "eyesight" is limited due to a modest radar transmission range, a vehicle can "see" surrounding vehicles and "hear" reports of their GPS coordinates.

In accord with other works, we assume that the majority of cars (about 85%) are honest [109]. By comparing what is seen to what has been heard, a vehicle can corroborate the real location of neighbors and isolate malicious vehicles to achieve local security. We expect the on-board radar device to provide useful corroboration of reported location information, except for during short transient periods. For example, the line-of-sight that radar needs may be temporarily obstructed by a large truck. Due to the dynamic nature of traffic, even if there are transient obstructions, the line of sight will be restored eventually.

Since radar has a range limit and the transceivers are limited by DSRC, we build network cells as security units as well as communication units. We use preset location-based cells as a basis for our approach. Each vehicle in the cell can directly communicate with every other vehicle in the cell. To achieve intra-cell security, a vehicle may use its radar to verify its neighbors' locations or issue queries to verify the location of a specified vehicle in the cell. In this way, each vehicle in the cell knows the location of the other vehicles in the cell with high certainty.

In addition, we propose a solution to provide inter-cell location information integrity for two reasons. The first reason is that applications often refer to location information of remote vehicles which are beyond a cell (ranging up to several miles). The second reason is that the on-board radar is not strong enough to verify vehicles in remote cells. When a vehicle receives an aggregated message which declares the

location of remote vehicles, the vehicle can randomly challenge and confirm the location of a vehicle in a remote cell by using the on-board radar in oncoming traffic. In this solution, we only cover the solution that challenges and confirms the location of a remote vehicle with the help of radar. When radar is not available, we can rely on location reports from the oncoming traffic or confirmed location reports from neighbors.

IV.2.1 Intra-cell Integrity

In this section we propose a method of verifying the location inside a cell, i.e. *intra-cell integrity*. Radar will be used to verify the reported location of vehicles, i.e. GPS location. Using radar, we can obtain the relative velocity, angle and location of the target vehicle. Both GPS location and radar location detections have measurement errors. The measurement errors will form a set of possible locations. If we can find an overlap between the two sets of locations from GPS measurement and radar detection, we can confirm the reported GPS location using radar detection.

GPS Location

In GPS, when satellite radio signals are transmitted, they are distorted by the troposphere and the ionosphere. Therefore GPS coordinates have some error. GPS data normally varies in the range of $\Delta x = \pm 10\text{m}$; $\Delta y = \pm 10\text{m}$ [110]. In Figure 27, we assume that Δx and Δy are always equal, marked as $\Delta x = \Delta y = \Delta\alpha$. The shadowed region is the set of possible real vehicle locations. We use (x, y) to represent the real location of vehicle. We can use (50) to describe this region.

$$(x - x_{gps})^2 + (y - y_{gps})^2 \leq (\Delta\alpha)^2 \tag{50}$$

Radar-Detected Location

Since radar also suffers from measurement errors, we assume that the radar's error includes two parts: angle error $\triangle\theta$ and radius error $\triangle\gamma$, marked as $(\triangle\theta, \triangle\gamma)$. In Figure 28, the region bounded by *HGQFEP* is the set of possible locations of the detected vehicle. We use (x, y) to represent the real location of vehicle in the GPS system and mark the radar readings as (θ, γ). We can use (51) and (52) to describe

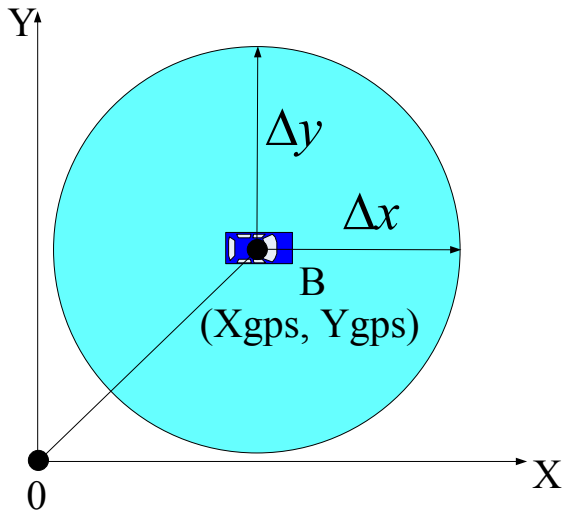


FIG. 27: GPS coordinates location. The GPS error results in a set of possible GPS location, shown as a shadow. $(x_{gps}, y_{gps})$is detection value of the GPS coordinates.

the circle D and circle C in Figure 28, respectively.

$$(x - \gamma \times \cos(\theta - \Delta\theta))^2 + (y - \gamma \times \sin(\theta - \Delta\theta))^2 \leq (\Delta\gamma)^2 \tag{51}$$

$$(x - \gamma \times \cos(\theta + \Delta\theta))^2 + (y - \gamma \times \sin(\theta + \Delta\theta))^2 \leq (\Delta\gamma)^2 \tag{52}$$

Here, θ is the detected angle, starting from north 0 degree and γ is the detected radius in meters (distance between vehicle A and vehicle B).

We note that there are two small regions HRG and EBF where the vehicle could be located, but these regions are not described by (51) or (52). Therefore we use the following formula to describe the region $FCGHDE$ in Figure 28:

$$\begin{cases} \gamma - \Delta\gamma \leq \sqrt{x^2 + y^2} \leq \gamma + \Delta\gamma \\ \theta - \Delta\theta \leq \arctan\frac{x}{y} \leq \theta + \Delta\theta \end{cases} \tag{53}$$

Although (53) includes some regions which are described by (51) and (52), for example the region $RGCFB$, this has no negative effect because we will find an intersection between the GPS location formula and radar location formula by using the technique to be described in the next section.

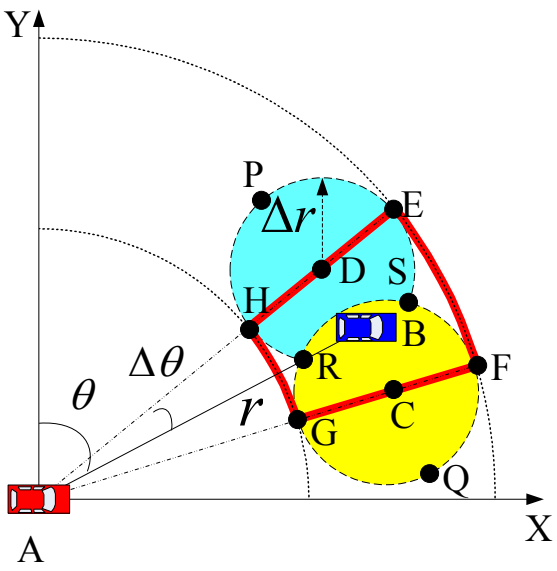


FIG. 28: Radar detected location.

Combining GPS and Radar Coordinates

To draw a conclusion, such as "*my neighbor is lying to me about its location*", we need to reason about the overlap (solution) between the GPS location formula and the radar location formula.

Without loss of generality, we assume that the possible vehicle is at the center of GPS location and radar location, shown as the lightly shaded area in Figure 29. If any of the following combinations has a solution, we can draw a conclusion that the detected vehicle is honest: (50) and (51); (50) and (52); (50) and (53). Otherwise the vehicle's location is infeasible and we conclude that the vehicle lies about its actual location. The meaning of these combinations is shown in Figure 29. If the GPS possible location intersects the radar possible location region, i.e. if there is an intersection between the GPS location shadow and radar location shadow, this means the GPS possible location is very close the value which is detected by the radar system. Therefore, we claim that we can accept the GPS location, i.e. we achieve the intra-cell location information integrity.

IV.2.2 Inter-cell Integrity

Similar to the idea behind greedy algorithms, local security is the basis of global security. In a greedy algorithm, we determine local optimal solutions and combine

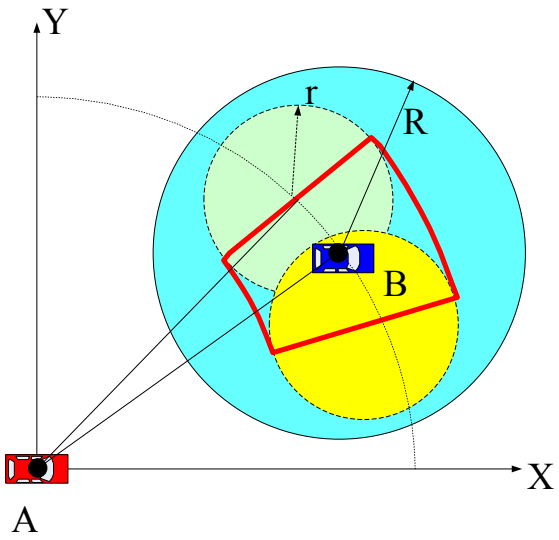


FIG. 29: Confirming GPS coordinates on GPS and radar location, if there is an intersection area between GPS location and radar location, we accept the GPS coordinates, otherwise discard it.

them to obtain a possibly optimal solution in the global region. Here, we use radar to obtain local security and combine the local security in various regions to get global security.

The basic idea of inter-cell security is to challenge and confirm the location of a vehicle in a remote cell by using the on-board radar in oncoming traffic. It is based on the fact that the vehicles in the same cell would see and hear almost the same traffic and road situation, so any modification done by malicious nodes can be detected by other honest vehicles. These honest vehicles then broadcast the correct record and isolate the malicious vehicles. An adversary may launch a Sybil attack or two types of location attacks: 1) the compromised vehicle continually lies about its location, 2) the compromised vehicle occasionally lies about its location. We can successfully solve the first type of location attack using the solutions described here.

Locally secured location needs to be propagated so that other vehicles approaching the cell may benefit from it. For this purpose, we use a cell router in each direction to minimize collisions and bandwidth usage. An alternative approach could be to use the distance from the message source as a factor in determining the next broadcast time. The reason we opted not to use it is because of the fact that in heavy traffic situations, there can be more than one car at approximately the same distance

(taking precision error into account). This would increase the chance of collisions. Other vehicles in the cell have the responsibility of monitoring the responses of the cell router or cell leader. If either of them tries to change the records or inject wrong information, honest vehicles can notify other members and initiate the process to find a cell router or cell leader and broadcast the correct record. There could be a case where the presence of many compromised vehicles might isolate honest vehicles. We propose a challenging method on the basis of the following facts: (1) it is most likely that majority of the nodes are honest (one of our assumptions), (2) even if in one cell there are more compromised than honest nodes, it is difficult to maintain such a topology in a VANET, and (3) to confirm whether the cell router is compromised, vehicles in other cells who have received records from this cell router can run a simple verification test as described below.

A message challenging method is used to verify records which are disputed. Whenever vehicles in a cell do not agree with the message broadcasted, they express their concern by broadcasting their records about the observed vehicle. Each record is associated with vehicle's ID (e.g. Electronic License Plate). A vehicle getting different information for the same vehicle can verify the disputed record through this challenging method.

Since we assume that traffic moves in both directions, a node can send the verification request to the cell router in the oncoming direction. We show an example in Figure 30. Traffic is moving in two directions: the bottom road is eastbound whereas the top road is westbound. Suppose car F transmits a message which is propagated backwards and eventually received by car A. If car A would like to verify the location received from car F, it can send the verification request using cars moving in the oncoming direction. Cell routers propagate the message to neighboring cells until it reaches the destination cell as determined by sender. The destination can be more than one cell depending on the location of the node when the record was sent, its speed and the cell where it was present. Once the car with the same ID is identified, its location can be verified and a response can be generated. If there is no such node in the potential destination cells, or the location information was modified, then the record is considered to be spurious and is dropped. The sender of that record comes under the scrutiny of other honest nodes.

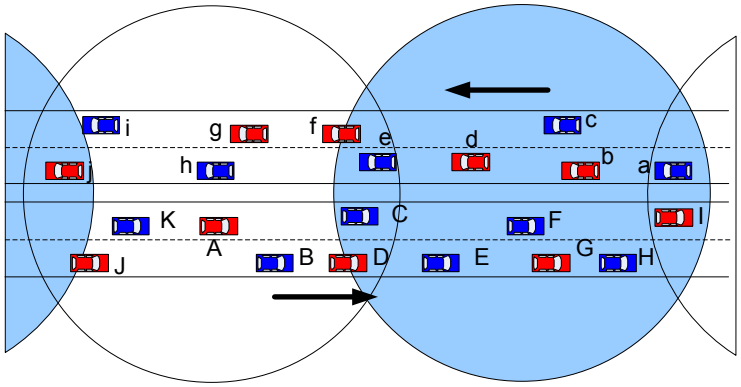


FIG. 30: Message routing among the cells.

IV.3 PASSIVE LOCATION INTEGRITY

For three major reasons, we propose a solution to validate location information without the help of radar. First, radar will not work when the line of sight is blocked, although the active location integrity model can prevent many location related attacks. In this scenario, vehicles have to trust their neighbors. But there definitely is a certain risk that the neighbors are the location attackers. Another problem in the active location integrity model is that there is no relationship between identity and the location of a vehicle. Suppose two vehicles a and b are at location p_a and p_b. An attacker announces that vehicle a is at location p_b and b at p_a. By using radar, we cannot tell if the statement is true or not because radar can only detect the presence of vehicles. No identities are related to the presence of vehicles. The third, not all vehicles may have an on-board radar installed.

Therefore, we propose a *passive location integrity* model. In this solution, we remove the requirement of a radar device but keep GPS and transceiver devices. Vehicles collect location information from neighbors and from the oncoming traffic. The collection of locations include reports from honest vehicles and reports from malicious attackers. We apply a statistical method to filter out malicious reports. After validating and filtering out false or inaccurate locations, we can refine the remaining location reports and obtain an average estimation of location.

The passive location integrity model provides intra-cell location information integrity. For inter-cell location integrity, we can keep the same algorithm as the one

presented in Section IV.2.1. We rely on the intra-cell location validation method to check the announced location. In inter-cell algorithm, we apply a suitable intra-cell location validation method in the last cell that includes the announced location. Therefore, the substitution of intra-cell method does not affect the inter-cell location integrity.

IV.3.1 Data Sources

Since radar is no longer present, we can only rely on the transceiver to obtain location information about other vehicles. When an observer receives an announced location from a neighbor, the observer monitors the source location of the neighbor by wireless signal strength. If the source location of the radio matches the announced location (with a certain error tolerance), the observer accepts the announced location. Otherwise, it ignores the announced location.

Based on the sources of location information, we can have two types of data: data from neighboring vehicles and data from oncoming traffic attackers. The location reports can come from neighboring vehicles shown in Figure 31. The neighboring vehicles are co-directional vehicles on the co-direction in a cell. The oncoming vehicles are ones on opposite direction in a cell.

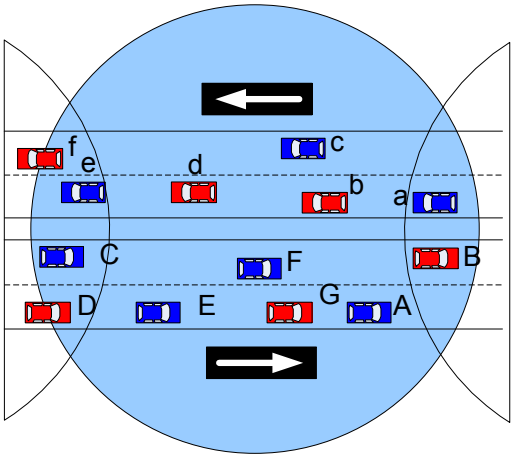


FIG. 31: Data sources. The neighboring vehicles are ones with upper cases letters. The oncoming traffic vehicles are one with lower case letters.

Based on the threat model, we assume that the location information collected by the observer includes two types of location reports. One type of location report is the one from malicious attackers, called malicious report. Since attackers can modify the

values of locations into any arbitrary number, there is no universal distribution of the location from malicious attackers. Figure 32 shows malicious locations as the squares deviating far from normally distributed location. Another type of location reports is the observation from honest vehicles. These observations generally have a consistent distribution, e.g. a certain error which is often assumed as a normal distribution. Figure 33 shows the one dimensional location with normal distribution. In reality, the collected location report depends on the number of neighboring vehicles. The mean value of this type of location report is close to the actual location. We assume that the majority of vehicles are honest. In this section, our motivation is to filter the malicious location and then to determine the accepted location from the collected location report.

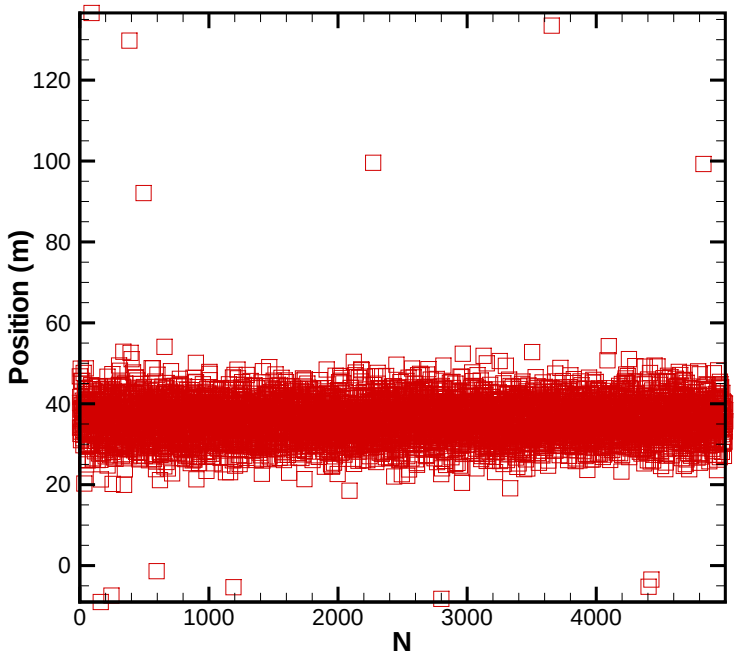


FIG. 32: Passive integrity: raw location reports. One square represents one location report. The outliers are malicious location report with abnormal values.

IV.3.2 Filtering Out Malicious Data

Once the location announcements/reports are collected, a statistical method will be applied to filter out the abnormal location reports sent by malicious vehicles. The statistical method is called *Mahalanobis Distance*, a distance measure introduced by P. C. Mahalanobis [111].

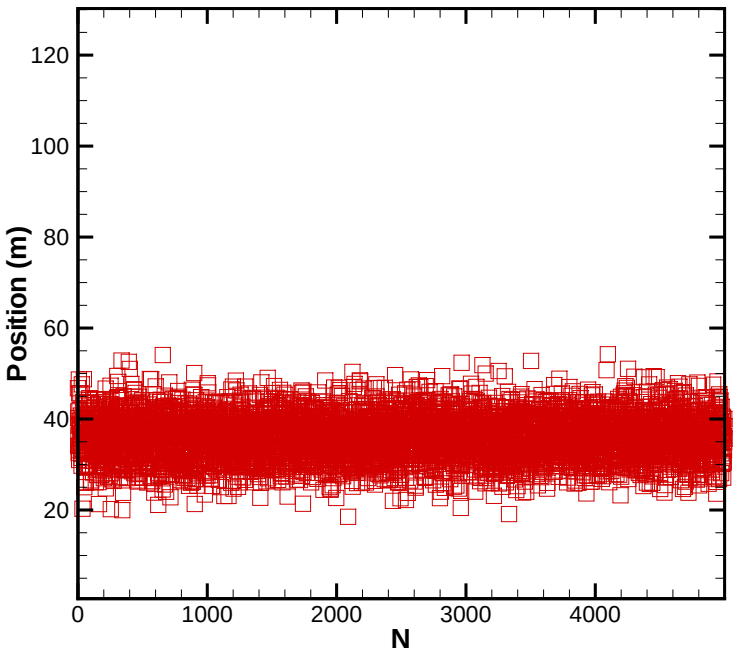


FIG. 33: Passive integrity: filtered location reports. One square represents one location report. These reports include measurement errors and are assumed as normally distributed.

The Standard Mahalanobis Distance

The Mahalanobis distance is a useful technique of determining the similarity between an unknown sample set and a known one. The idea of Mahalanobis distance is the following: if two input locations belong to the same distribution, the Mahalanobis distance value of the two locations will fall into a certain range; on the other hand, if two input locations do not belong to the same distribution, the Mahalanobis distance value of the two locations will be far from the range. Base on this fact, we can filter out the abnormal locations. Formally, the Mahalanobis distance is defined as a dissimilarity measure between two random vectors $\vec{x}$ and $\vec{y}$ with the same distribution and covariance matrix S :

$$d(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T S^{-1} (\vec{x} - \vec{y})}.$$

If the covariance matrix is the identity matrix, the Mahalanobis distance reduces to the Euclidean distance which is the distance between two points, i.e.

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}.$$

We assume the sample points are distributed about the center of mass in a elliptical manner.

An Intuitive Explanation

It is known that the normally distributed data can be illustrated using a ellipse in a two dimensional space. As an intuitive explanation, the Mahalanobis distance is simply the distance of the test point from the center of mass divided by the width of the ellipse in the direction of the test point. For example, A and B are two test points, shown in Figure 34. The Mahalanobis distance value of A is $\frac{OA}{OW}$ where OA is the distance of the test point A from the center of mass O and OW is the width of the ellipse in the direction of the test point A. The Mahalanobis distance value of B is $\frac{OB}{OW}$ where OB is the distance of the test point B from the center of mass O and OW is the width of the ellipse in the direction of the test point B. If the Mahalanobis distance of a test point is large, the test point must be a outlier which will be abandoned from the sample set. In our problem, the attackers will give an abnormal location report which is far away from the normal location reports. By applying the Mahalanobis distance, we can filter out these malicious reports from the sample set. For example, OB has a larger value and is larger than 1. Therefore, B must be an outlier. In three dimensional space, the Mahalanobis distance has a similar definition. For example, test points A and B have the Mahalanobis distance value $\frac{OA}{OW}$ and $\frac{OB}{OW}$ respectively, shown in Figure 35.

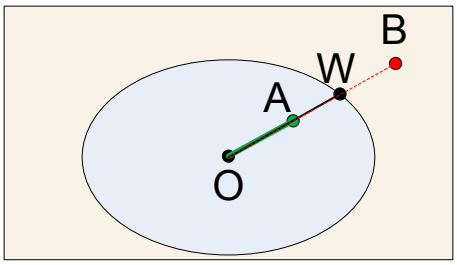


FIG. 34: Two-dimensional space.

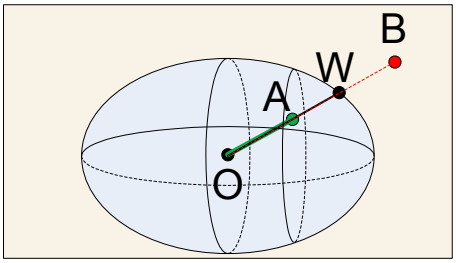


FIG. 35: Three-dimensional space.

The Problem

Based on the data source introduced earlier, the location information collected by an observer includes two types of data: good reports and malicious reports. The good reports are from honest vehicles, and the malicious reports are from malicious attackers. Based on our assumption, we assume that the malicious reports are locations with abnormal values, i.e. outliers. For each of the n observations in a p-dimensional data set, a Mahalanobis distance value MD_i is calculated. Let $\overline{x}$ be the sample mean vector and let V be the sample covariance matrix,

$$V = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T. \tag{54}$$

Then

$$MD_i = \sqrt{\{(x_i - \overline{x})V^{-1}(x_i - \overline{x})^T\}} \tag{55}$$

for $i = 1, \cdots, n$. Observations with large D_i-values may be considered to be outlying.

D_i is composed by the mean $\overline{x}$ and the sample covariance V. Traditionally, the sample mean and the sample covariance matrix give a good estimation of data location and data shape if it is not contaminated by outliers. When the location input is contaminated, the mean $\overline{x}$ and the sample covariance V will deviate and be significantly affected by the outliers. Therefore the value of the Mahalanobis distance affected by outliers will deviate from the real value as well and result in a false location estimation which will harm location integrity.

A New Mahalanobis Distance

To remove the deviation of outliers, we replace the sample mean $\overline{x}$ and the sample covariance V by the sample median x^* and the robust covariance S, respectively. The robust covariance S is defined as follows:

$$S = \frac{\sum_{i=1}^{n} K(\|x_i - x^*\|)(x_i - x^*)(x_i - x^*)^T}{\sum_{i=1}^{n} K(\|x_i - x^*\|)}, \quad (56)$$

where $\|X\| = XV^{-1}X^T$, and $K(u) = \exp(-hu)$. As recommended by Caussinus and Ruiz [112], $h = 0.1$. After replacing the sample mean $\overline{x}$ and the sample covariance V, the new Mahalanobis distance D_i^r is calculated as

$$D_i^r = \sqrt{\{(x_i - x^*)S^{-1}(x_i - x^*)^T\}} \quad (57)$$

D_i^r excludes the deviation caused by the outliers by the following rules. For each input location, we calculate the value of D_i^r. For multivariate normally distributed data, the values of D_i^r are approximately chi-square distributed with p degrees of freedom(χ_p^2) where p is the dimension of the ordinal data, i.e. $p = 2$ [113]. The observations with large D_i^r values can be abandoned by using a quantile of the chi-squared distribution (e.g., the 97.5% quantile). Once the outliers are expelled, we can obtain the location estimation by takeing the mean value of the remaining location reports.

IV.3.3 Improving Location Resolution

In reality, the sample distribution may not be close to the underlying distribution (normal distribution). To get a more accurate estimate and construct a confidence interval for the estimate, we use a resampling method called *bootstrapping*. The bootstrapping method allows one to gather many alternative versions of the single statistic that would ordinarily be calculated from one sample with replacement. This method has the advantage that one can compute the high resolution estimation when the sample size is small or the population distribution is unknown. By resampling, the mean is repeatedly updated and refined and the precision can be improved. The bootstrap estimation procedure consists of the following steps:

1. Use the original data set $(x_1, x_2, x_3, x_4, \cdots, x_n)$ to calculate the mean. Call this $\overline{x}_0^*$, where 0 represents the initial step.

2. Take a set of bootstrap samples of size n from the original data set with replacement, which produces a new data set: $(x_1^*, x_2^*, x_3^*, x_4^*, \cdots, x_n^*)$. Calculate the mean of the bootstrap samples. Note: Some of the x's will be duplicate or triplicate observations of the same value. Call this $\overline{x}_1^*$.

3. Repeat step 2 N times, where N is a large number. This produces $\overline{x}_1^*, \overline{x}_2^*, \cdots, \overline{x}_N^*$.

4. Finally, the bootstrap estimate of mean can be calculated as

$$\overline{x}^* = \frac{\sum_{k=1}^{N} \overline{x}_i^*}{N} \tag{58}$$

The final estimate of location can be obtained from the remaining reports by computing the mean value of the locations. The mean value of the remaining report is acceptable because abnormal values of reports have been filtered by the Mahalanobis distance. We can compute a parametric bootstrap confidence interval using the $\overline{x}_i^*$s. The standard deviation of the $\overline{x}_i^*$s, i.e. SE is the estimate of the standard error of $\overline{x}$. An approximate confidence interval is $\overline{x}^* \mp z_{\frac{\alpha}{2}} SE$, where $z_{\frac{\alpha}{2}}$ is the critical value of the normal distribution.

IV.4 GENERAL LOCATION INFORMATION INTEGRITY

In real environments, some vehicles will have on-board GPS device and a transceiver, some will have transceiver only, and some older vehicles will not have eithor of these. In this section, we present a novel solution to obtain the location coordinates among the collected input from a noisy environment, i.e. from a real environment. The collected location reports as input samples are filtered by the Mahalanobis distance. After filtering, the inputs from different resources are weighted and averaged to obtain the location estimation which is with a certain acceptable precision. The contributions of this section are the following: 1) filtering out false location reports from malicious attackers; 2) computing high-resolution location information from anonymous low-resolution location observations; 3) obtaining an efficient filtering algorithm.

IV.4.1 Data Sources

The location samples as statistical input come from three types of resources: the radar device of the observer vehicle, the oncoming vehicles, and the neighboring vehicles of the observer vehicles. The location reports from the radar device are assumed to be a valid location estimation. The neighboring vehicles of the observer vehicles have the highest risk to be contaminated. The precision of oncoming vehicle reports are considered to be in between the precision of radar and the precision of neighboring vehicles. In Figures 36 and 37, squares represent data from neighboring vehicles' detection. Squares represent location collected from the neighboring vehicles. The small circles represent data collected from oncoming vehicles, and the small triangles represent data from radar detections.

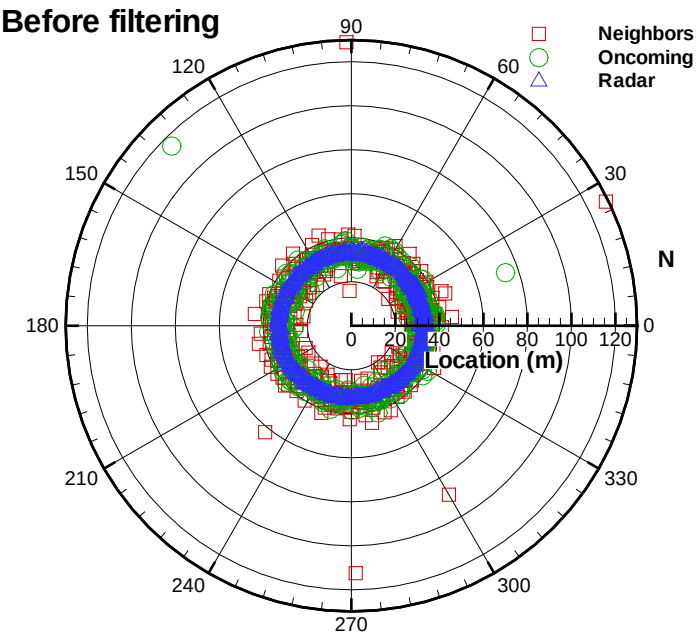


FIG. 36: Collected raw locations including malicious outliers. N is the number of locations. The outliers (malicious data) are far away from the center.

We can apply the filtering method of Mahalanobis distance to filter out the malicious data in each data source. Therefore, we can obtain a set of reported locations which only include measurement errors. These location reports are ready to refine to compute the final estimation of location.

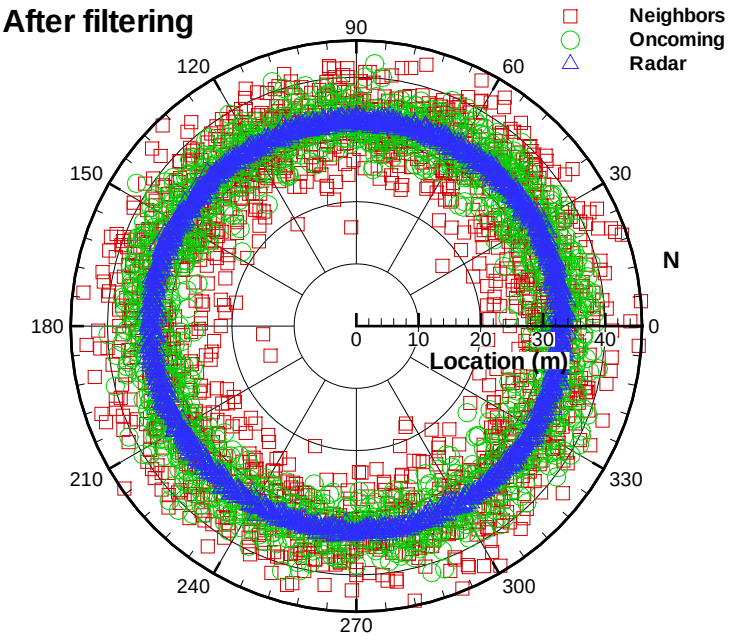


FIG. 37: Gaussian error location. N is the number of locations. One symbol represents one location. The center point is the actual location of the observed vehicle.

IV.4.2 Filtering Out Malicious Data

There are three sets of location data that can be obtained from three different sources. Each set of location data can have outliers from malicious attackers and normally distributed data from measurement errors. The basic idea of filtering the malicious data from the collected data is by computing the new Mahalanobis distance.

Let **X**, **Y**, and **Z** represent the location detection variables from radar detection, oncoming detection, and neighbor detection, respectively. Suppose we have **L**, **M**, **N** samples of radar detection, oncoming detection, and neighbor detection, respectively. By applying the filtering method discussed in Section IV.3.2, we can obtain the mean of each data source. The final estimation of the location can be written as

$$P = w_1 * \overline{x}^* + w_2 * \overline{y}^* + w_3 * \overline{z}^* \tag{59}$$

where w_1, w_2, w_3 are the weights of radar detection, oncoming detection, and neighbor detection, respectively. $\overline{x}^*, \overline{y}^*, \overline{z}^*$ are the estimated mean by filter method in Section IV.4.2.

IV.5 ISOLATING MALICIOUS VEHICLES

This section presents a scheme that can be used to manage vehicles, for example isolating a malicious vehicle. Each vehicle is represented by a piece of memory. The piece of memory has a fixed size. The design of the memory must be efficient because we will record as many vehicles as possible. A vehicle ID is used to identify the memory. The content of the memory includes the mobility history and the trust status. A vehicle in memory is shown in Figure 38. The mobility history records the mobility information, i.e. location, speed, acceleration and time. The mobility history or *Map History* can be used to validate location announcement to improve location integrity. Each vehicle has a trust status, *trusted*, *questionable*, or *suspect* which is managed by tables in memory. We can isolate malicious vehicles by putting the vehicles into the suspect table. All messages related to vehicles in the suspect table will be directly dropped. Each new vehicle is put in the questionable table. All messages related to vehicles in the questionable table will be randomly checked. Good behaving vehicles are put in the trusted table. All the messages to and from vehicles in the trusted table will be directly propagated.

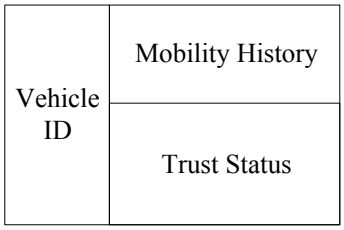


FIG. 38: A vehicle in memory. The mobility history records the mobility information of the vehicle. The trust status records the trust information of the vehicle.

Since the cell leaders and the cell routers are special vehicles, we can specially design a mechanism to secure these special vehicles. The mechanism is on the basis of an assumption: the majority of vehicles are honest.

IV.5.1 Map History

The mobility history of a vehicle can be stored in memory. The purpose of this history is to validate "real" data from "fabricated" data. The basic idea is that any vehicle without historical consistency is highly suspect. All location information must be consistent with the location history.

Location History Levels

We are interested to store the mobility history of a vehicle. The information stored includes location coordinates, time, mobility parameters, identity and time. The memory of each vehicle is limited and we want to record vehicles in an intelligent way that can covers longer location history in a limited memory space.

To save memory space, we record location history at a certain rate e.g. 1s, 10s, 30s, 60s, 5min, etc. One second records will form a set of locations whose interval is one second. Ten second records will form a set of location whose interval is ten seconds, and so on. We can have N sets of location records with N intervals. We term each set of location records as a *Level* as shown in Figure 39. The time interval that is used to build a level is called *Level Interval* $INT_i, i = 0, 1, 2, \cdots, N$ and $INT_0 < INT_1 < \cdots < INT_N$. For example, $INT_1 = 1s$ for the Level One. Level One will have a set of locations whose interval is one second. $INT_2 = 10s$ for the Level Two. The Level Two will have a set of locations whose interval is ten seconds.

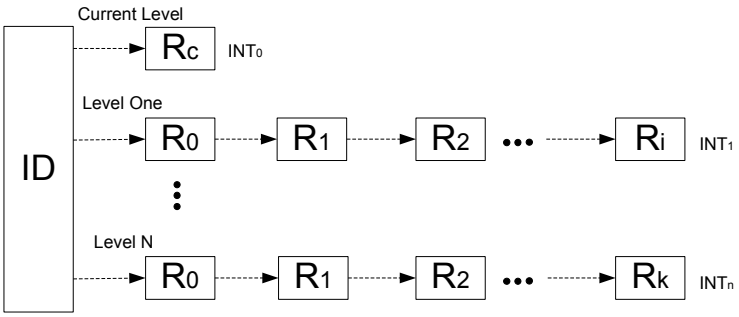


FIG. 39: A vehicle in memory with levels. R_i is the record of mobility information. The level with smaller level number has more detailed records and shorter intervals.

A more direct overview of the map history is shown in Figure 40. Level One represents locations recorded in a rate of one record per second, i.e. $INT_1 = 1$. In one second, a vehicle moves at most 33 meters if we assume the maximum speed is 33 m/s (75 mph). Therefore, the interval between t_i and t_{i+1} is at most 33 meters. Level Two includes records which are collected every 10 seconds, i.e. $INT_2 = 10$. The interval between t_i and t_{i+1} is at most 330 meters. Suppose for each level, we have 10 records. If we want to validate a vehicle with a location 110m from its reference location (i.e. t_0), we can use Level One because 110m is less than the maximum

location of Level One 33*10 or 330m. If we want to validate a vehicle with a location 689m from its reference location, we will use Level Two because 689m is larger than the maximum location of Level One 33*10 or 330m but is less than the maximum location of Level Two 330*10 or 3300m.

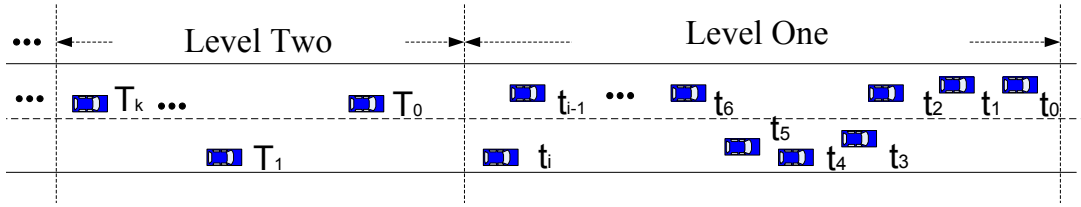


FIG. 40: Map history overview. Level One has one location record per second. Level Two has a one location record per 10 seconds.

Validating Location by Map History

Once the map history is created, we can distinguish the "real" locations from the "fabricated" ones. The idea of validation is shown in Figure 41. In level i, a valid location must be in the area which is bounded by the predicted locations and the roadways. For example, the locations between time t_2 and t_3 must fall into the shaded region, ABCD. All the locations between time t_2 and t_3 but not in this region are considered to be fabricated.

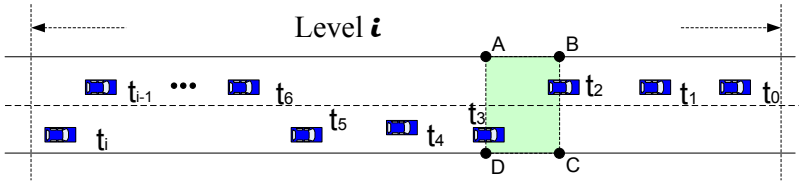


FIG. 41: Screening vehicles using the map history. A vehicle with location between time t_2 and t_3 must fall into the shaded region, ABCD. Otherwise it is a suspected vehicle.

We use time, mobility information, and location to validate locations. Subsequent

vehicle locations can be described by the following formula:

$$\begin{cases} (x - x_i)^2 + (y - y_i)^2 \leq \gamma^2 \\ \gamma = t * \nu_i \\ 0 \leq \nu \leq v_{max} \\ \{x, y\} \in Road \end{cases} \tag{60}$$

For any Level i, we have a predicted record t_0 which is given based on the maximum velocity:

$$\begin{cases} (x - x_i)^2 + (y - y_i)^2 \leq (t * v_{max})^2 \\ \{x, y\} \in Road \end{cases} \tag{61}$$

In (60) and (61), x_i, y_i are the coordinates of record i in any Level N, x, y are the coordinates of new data or record, t is the length of sample interval of Level N, and v_{max} is the speed limit on the road. We can consult the digital map with the (x, y) coordinates to check if (x, y) is on the road. So far, we can screen all the data by solving (60) and (61). The physical meaning is that if the new data is inside the shaded region in Figure 41, we can declare the data acceptable. Otherwise it is not acceptable, and the vehicle will be isolated by the method in Section IV.5.2.

Validating Examples

The map history can be used to validate locations. To illustrate this, we zoom in one level and show the detail of one level. Locations of the vehicle in history are stored in the level. Each location is associated with a time $t_j, j = 0, 1, \cdots, i$. We give a detailed level example in Figure 42. A location between t_2 and t_3 must be in the region bounded by the roadways and the locations t_2 and t_3. If the vehicle announces that a location between t_2 and t_3 is at A, we can reject the location because A is out of the boundary. In another example, if the vehicle announces that a location between t_1 and t_2 is at B, we accept this location because the location B is in the region formed by the roadways and the locations t_1 and t_2. If a vehicle has just joined the traffic, we will give this vehicle a period of time to build a history.

IV.5.2 Trust State of Vehicles

We use the trust states *Trusted*, *Questionable* and *Suspect* to isolate malicious vehicles. All vehicles in a cell maintain the trust status in memory as shown in Figure

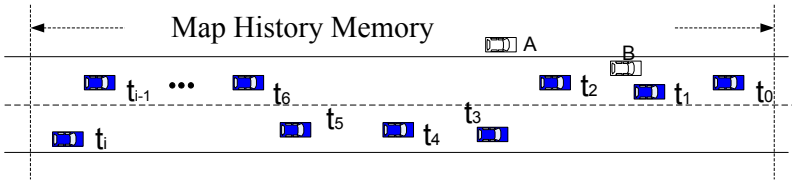


FIG. 42: Map history example. A is an impossible location because it is outside the road; B is an incorrect location because it is supposed to be between t_0 and t_1, where t_0 is the predicted location and t_1 is the last received location.

38. The trust status includes three tables, the trusted table, the questionable table and the suspect table, as shown in Figure 43. We will address the initial state and the transition of the trust status of a vehicle.

Initially, a new vehicle enters the road, as an observer. The new vehicle enters a cell and broadcasts a "HELLO" message with its location information. The cell leader will place the vehicle into the *questionable table*. The cell leader then sends back the cell leader's ID, the cell routers' IDs and all the map history of the cell. When the leader sends information to the observer, members in the cell will hear it. If there is bogus information in the message, a member may dispute the leader. The new vehicle places the cell leader and routers into its *trusted table* and sets up the initial map history received from the cell leader. From then on, the new vehicle starts to build its own map history.

The trust status can be changed on the basis of the behavior of the vehicle. Figure 43 shows the tables and state transitions. If an observed vehicle behaves normally for a certain period of time, the observer will place the vehicle into the *trusted table*. If an observed vehicle behaves abnormally, the observer will place the vehicle into the *suspect table*. Otherwise the vehicle will stay in the current table. Here normal behavior means that the reported location has been validated by methods presented earlier. There is a timeout rule for a vehicle. If not receiving reports from an observed vehicle or not detecting the observed vehicle for a while, the observer will move the observed vehicle from the trusted table to the questionable table. The transitions between states are based on "Credit Score". A vehicle will obtain one credit score if the vehicle behaves normally, for example sending 100 correct location messages. The vehicle will quickly lose one credit score if the vehicle behaves abnormally, for

example sending one abnormal location message. We give each newcomer an initial score: 3. The minimal score for being in the trusted state is 5. The score for being in the suspected state is 0.

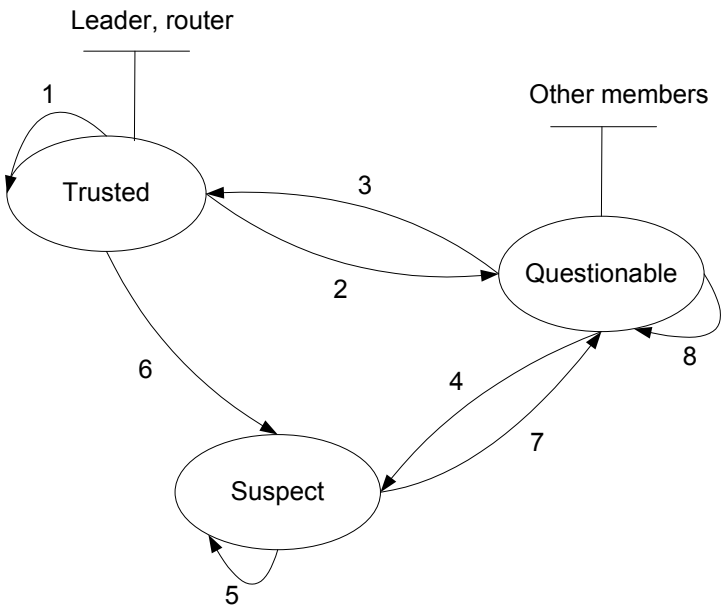


FIG. 43: State transitions. State transitions 1, 3, 5, 7: if confirmed. State transitions 2, 4, 6, 8: if not confirmed.

The states of vehicles can be synchronized in two ways: proactively and reactively. The proactive approach is based on the arrival of new vehicles in a cell. When a new vehicle enters the cell, the cell leader will broadcast the states of vehicles in the cell. Members in the cell can synchronize their state tables. The reactive way is based on a timer. A synchronization timer is preset for all the cell members. When the timer expires, the cell leader will broadcast the current state table to synchronize the tables.

IV.5.3 Securing Cell Leaders and Cell Routers

Cell leaders and cell routers are special vehicles elected by the cell members. Since the cell leaders and the cell routers represent all the members of a cell, corrupted cell leaders and cell routers can cause great harm. The basic idea of securing them is to monitor them using other cell members. Cell routers are monitored and challenged by comparing messages. When a cell router is sending or receiving packets, its neighbors

will monitor the packets. If the cell router modifies, inserts bogus information, or drops a packet, its neighbors will detect this by comparing the packet received from the cell router with the original packet since the neighbors receive same packet as the router does. If the neighbors detect that the router is compromised, they will isolate the router. The isolation of cell leaders or cell routers is done by listing vehicles into suspect tables which is discussed in the previous section IV.5.2.

The cell leader is monitored and challenged by other members of a cell as well. Since all vehicles are homogenous, they can generate the aggregated location packet and compare it with the one sent by cell leader. This prevents the cell leader from changing location information. If cell members keep silent, then it is an indication that they agree with the message. Otherwise, vehicles can challenge the cell leader. The challenge will finally be resolved by polling in the cell. The polling involves three rounds of messages. 1) One vehicle arises a challenge. 2) All the members reply with their own votes. Therefore, each vehicle will receive votes from other members. Vehicles in a cell can count votes to solve the challenge. 3) If the leader is honest, vehicles remaining silent agree with the current cell leader. Otherwise, vehicles will start a new election for the cell leader. The decision will follow the majority vehicles' opinion. The challenger will be punished by decreasing its credit score. Therefore, we have only two rounds of messages if the cell leader is honest and three rounds of messages if the cell leader is not honest. Since the population in a cell is normally not large, the time and control message costs of the polling algorithm are acceptable. Cell routers will propagate the corrected message to next cell. These steps will repeat until the destination vehicles receive the message.

IV.6 SUMMARY

In this chapter we have proposed solutions for location integrity which ensures that location information is original (from the generator), correct (not fabricated) and unmodified (value unchanged). We started from a simple system with a strong assumption that all vehicles are equipped with radar, GPS receiver and transceiver. An active location integrity algorithm is proposed. Radar acts as a virtual "eye" of a vehicle, and the transceiver acts as a virtual "ear" of the vehicle. The idea of active location integrity follows the adage: seeing is believing. When vehicles receive the announced location information from the virtual ear, the receiver will actively detect the announced location by using the virtual eye. We then weaken

the assumption by removing radar from the device list. Vehicles have only a virtual ear. We proposed the passive location integrity algorithm. When vehicles receive the announced location information from the virtual ear, we filter and refine the location announcement by computing similarity and applying the map history. We finally came to the real world solution in an environment that some vehicles have some combination of the devices (radar, GPS receiver and transceiver). In the real world solution, a statistical method was addressed and the location information can be filtered and refined. A weighted average of the location estimation can be computed after filtering by the statistical method.

CHAPTER V

LOCATION CONFIDENTIALITY

Location information from multiple vehicles is often aggregated to reduce the number of messages that are sent. In VANET transmission, plaintext location information, especially aggregated location information, can be used and stored because other vehicles need access to validate the location and monitor cell leaders and cell routers. Given the insecure nature of wireless communication, plaintext messages are vulnerable because an attacker can easily modify the location information and harm the location integrity. Therefore, we propose both encryption/decryption and access control mechanisms to provide location *confidentiality* in this chapter. In VANETs, most applications require protection of location information. In some environments, for example in the battlefield, the locations of vehicles need to be extremely secure. Therefore, not only the content is required to be encrypted but also the place where the encrypted message can be decrypted is required to be specified. Figure 44 shows that vehicle Alice specifies a decryption region (shaded area) and vehicle Bob must be physically present in the decryption region to decrypt the message. The decryption region can move along with Bob as well.

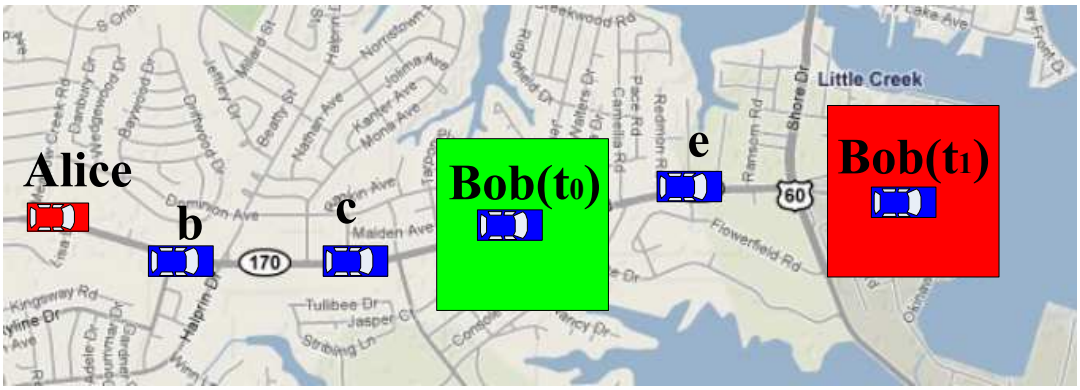


FIG. 44: Vehicle Bob must physically present in the shaded region to decrypt message. The shaded region (decryption region) can move along with Bob.

In our approach, a special region (decryption region) is specified for the message receiver. The receiver must be physically present in the decryption region to decrypt the received message. To achieve this, the receiver's location information is converted into part of a key. The message is encrypted using both the converted key and PKI.

The idea of using location as part of a key is called GeoEncryption and was first proposed by Denning *et al.* [79, 2]. But there are shortcomings to using GeoEncryption in VANETs, which will be discussed in Section V.1. Our main contributions in this chapter include:

- Designing an algorithm to convert a location into a key,
- Reducing the overhead in terms of control messages which update the decryption region,
- Improving location error tolerance.

V.1 DENNING'S GEOENCYPTION

To implement the idea shown in Figure 44, location information will be converted to a secret key. The idea of using location as part of key is first proposed by Denning *et al.* in GeoEncryption [79, 2]. Denning's GeoEncyption applies a table (GeoLock) to convert the location into a key. The GeoLock table is shown in Figure 45. All vehicles need to pre-install this table to convert the location into a key. Like updating passwords, we need to periodically update the GeoLock table to protect the secret keys. One issue is that the GeoLock table must be consistent on all vehicles. If the table on some vehicles is not up-to-date, it is impossible to decrypt the message. Given the large scale of vehicle population, it is very challenging to keep tables up-to-date. We propose a new algorithm to convert the location into the secret key by dynamically computing the key instead of using a table lookup.

Denning's GeoEncryption also requires that both sender and receiver are pre-installed with PKI. The message from a sender to a receiver needs PKI. The algorithm of GeoEncryption is shown in Figure 46. The sender obtains a random number Key_S. Key_S is used to encrypt a plaintext message. The ciphertext is propagated to the receiver. The sender generates a secret key (GeoLock) from the GeoLock table. The GeoLock is XOR-ed with Key_S and then encrypted using the receiver's public key Key_E to produce the ciphertext $E\{Key\}$. The ciphertext $E\{Key\}$ is propagated to the receiver. The receiver decrypts the ciphertext $E\{Key\}$ by using private key Key_D to produce an outcome. The outcome is XOR-ed with the GeoLock obtained from the GeoLock table, to uncover Key_S. Once Key_S is uncovered, Key_S can be used to decrypt the ciphertext and uncover the plaintext.

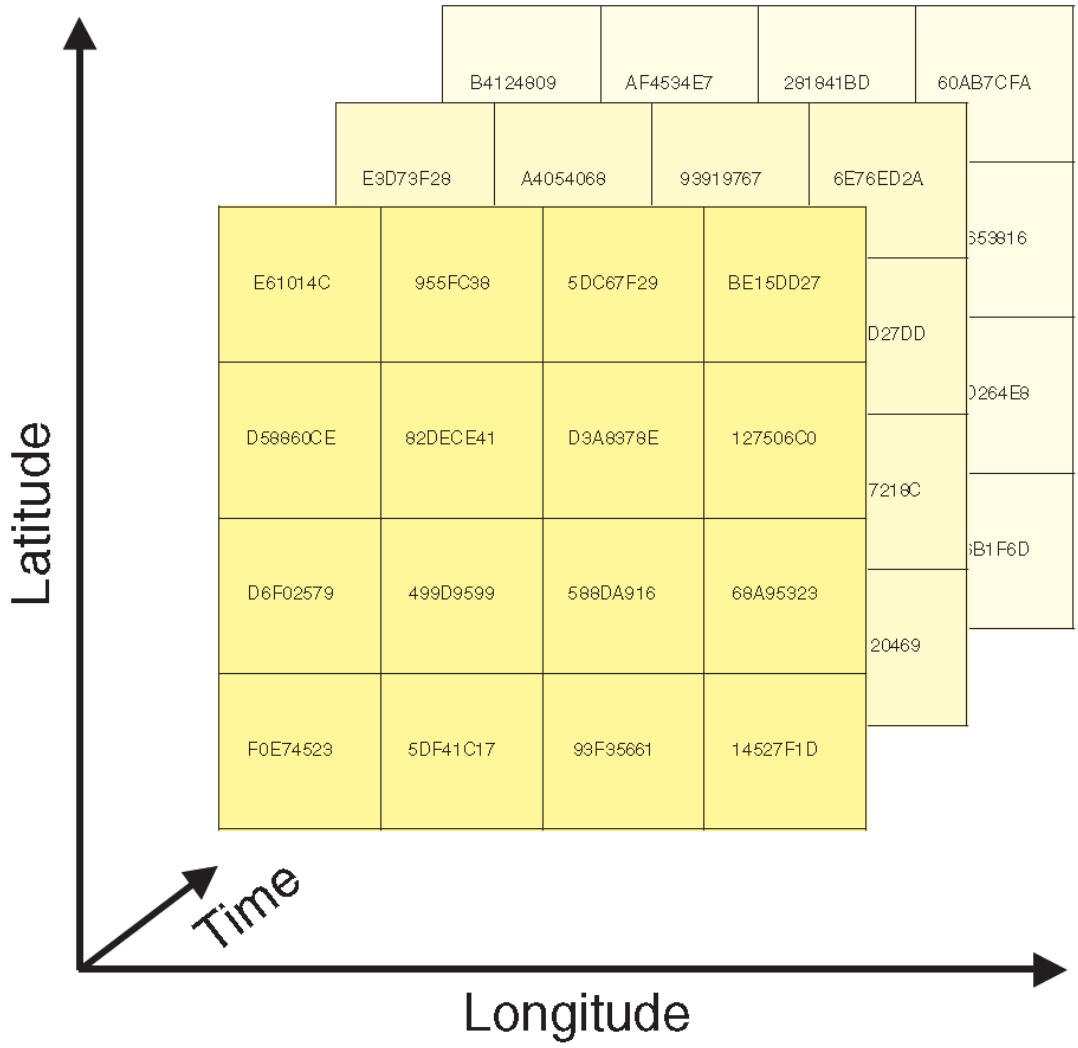


FIG. 45: Denning's GeoLock table [2] (Used with permission.)

Key_S can serve as session key for the communication of the sender and receiver. If attackers can record the initial handshaking message and location information of the receiver, the attacker can crack the Denning's GeoEncryption and obtain Key_S by using some hacking methods, such as dictionary lookup, hash-collision, social engineering, etc. Once Key_S is obtained, the information is at risk to be decrypted because a single and unique session key Key_S is used. Therefore, we propose a new algorithm that can update the session key and avoid the use of a single session key.

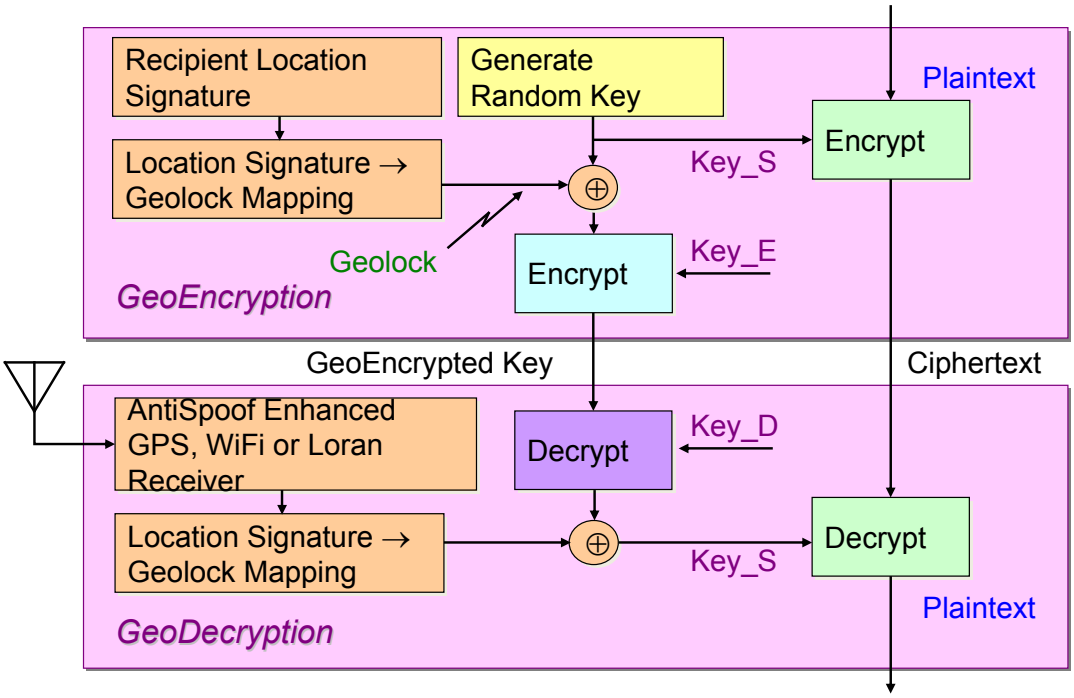


FIG. 46: Denning's GeoEncryption [2] (Used with permission.)

V.2 DYNAMIC GEOENCRYPTION

We extend Denning's GeoEncryption by proposing a new GeoLock that can dynamically convert the location into a secret key. Our dynamic GeoEncryption algorithm, called D-GeoEncryption, can dynamically encrypt and decrypt messages without checking tables. The new GeoLock, called D-GeoLock, is treated as a black box. The details of the black box will be covered in Section V.2.2. Figure 47 shows the device as a black box.

To avoid using a single session key, we propose a new GeoEncryption algorithm (i.e. D-GeoEncryption) that can dynamically change the session key. Our algorithm

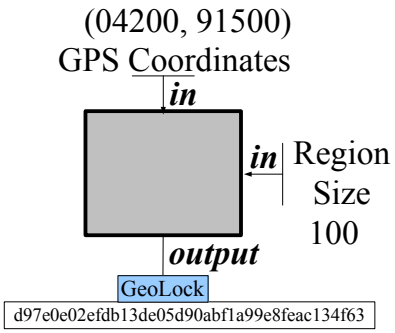


FIG. 47: An example of D-GeoLock. The input of region size is 100m. The GPS coordinates are (04200, 91500). The black box produces the D-GeoLock output.

involves a security key handshake stage and a message exchange stage, as shown in Figure 48.a. In the key handshake stage, the sender and the receiver negotiate a shared symmetric key. The sender will send a message which includes the sender's location information and mobility information. The sender generates two random numbers as keys, Key_S and Key_C. Key_C is appended to the original plaintext to generate a new message. The new message is encrypted by Key_S using a symmetric algorithm, e.g. Triple DES Encryption. This outcome of the encrypted message is $E\{Req\}$. The sender generates a D-GeoLock key based on the location of the receiver using the D-GeoLock algorithm which will be discussed later. Key_S is XOR-ed with the D-GeoLock key. The outcome of XOR is then encrypted using the receiver's public key Key_E to produce the ciphertext $E\{Key\}$. Both $E\{Req\}$ and $E\{Key\}$ are transmitted to the receiver through the wireless channel. When the receiver receives $E\{Key\}$, the receiver will decrypt $E\{Key\}$ to obtain the outcome of XOR using the receiver's private key Key_D. The D-GeoLock key is generated from the GPS location of the receiver is XOR-ed with the outcome from decryption and use to uncover the secret key Key_S. Once the Key_S is obtained, $E\{Req\}$ can be decrypted by using Key_S. Both the secret key Key_C and the original plaintext can now be obtained. Now Key_C is known by both the sender and the receiver.

In the message exchange stage, the receiver and the sender use the shared Key_C to communicate, as shown in Figure 48.b. When the receiver wants to reply to the sender, the receiver generates a random number Key_S'. The reply message is directly encrypted using Key_S' to generate a ciphertext $E\{Rep\}$. With the sender's location and mobility information, the receiver can predict the decryption region of

sender and generate the D-GeoLock key of the sender using D-GeoLock algorithm. The D-GeoLock key is XOR-ed with Key_S' and then encrypted by Key_C to generate a ciphertext $E\{Key'\}$. Both $E\{Rep\}$ and $E\{Key'\}$ are transmitted to the sender through the wireless channel. $E\{Key'\}$ is then decrypted using Key_C to recover the XOR outcome of the D-GeoLock key and Key_S'. The sender generates its own D-GeoLock key based on its current location. The D-GeoLock key is XOR-ed with the outcome to uncover the secret key Key_S'. $E\{Rep\}$ can be decrypted using Key_S', and the reply message is recovered as well. Now Key_S' is known by both sides. The sender can use Key_S' as a new secret key and repeats the algorithm to communicate with the receiver.

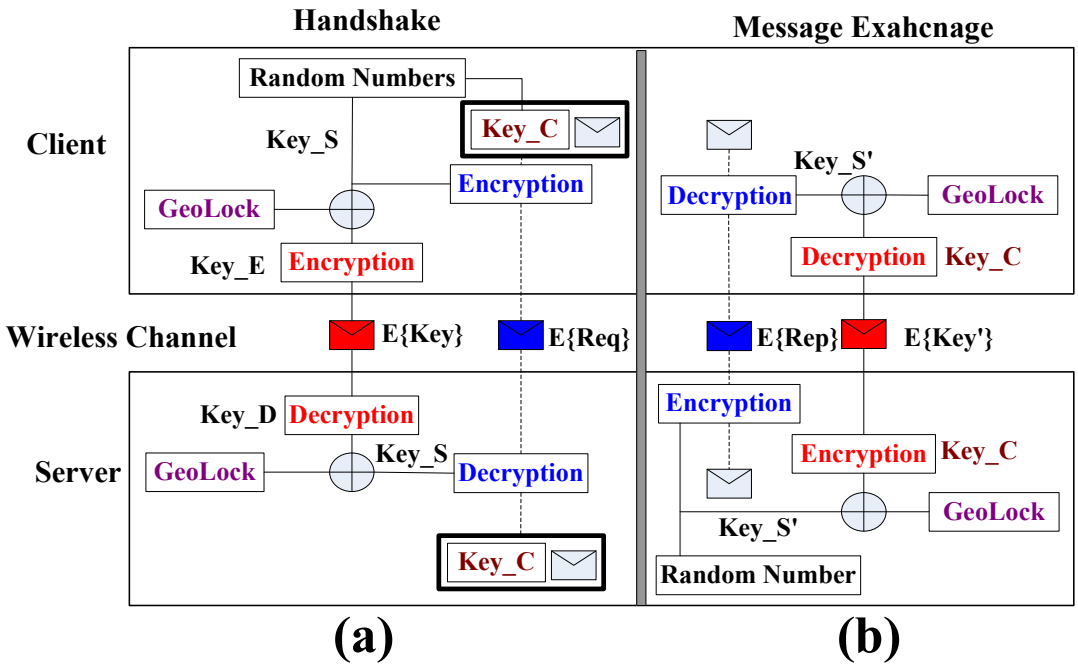


FIG. 48: Illustrating the proposed encryption and decryption scheme.

V.2.1 Decryption Region Prediction and Updating

The decryption region is a special area where messages can be decrypted. If the decryption region is larger, it will be more likely to cover some attackers. If the decryption region is small, the vehicles which will decrypt the messages will be more likely placed out of the region because of a location error. Therefore, the decryption region is specially specified to fit the targeted decryption vehicles.

Since vehicles can dynamically move, the decryption region needs to move along

with the vehicles who will decrypt messages. For example, vehicles in the battlefield will move. The decryption region of the vehicles is expected to move with these vehicles. Since vehicles have high mobility, we propose two improvements to determine the decryption region: predicting the decryption region and updating the decryption region. The movement of vehicles is constrained by roads, and the map of the roads can be accessed by all vehicles. Therefore, we can predict a vehicle's location based on the map and the vehicle's mobility. Based on the prediction of the decryption region, the messages are checked by geographic location. Because of the high mobility of vehicles, there will be a certain prediction error. Therefore the predicted decryption region will be corrected by updating the real location of the decryption vehicles. The real locations of the decryption vehicles are piggy-backed on other messages.

Prediction of the Decryption Region

For location coordinates we use Universal Transverse Mercator (UTM) [114]. GPS coordinates computed by the GPS receiver will be converted into UTM coordinates. One feature of UTM coordinates is the ability to provide a more precise location by simply adding a pair of digits to abbreviated coordinates. For example, an 8-digit UTM location is accurate to 10 meters, approximately the size of a house, and a 10-digit UTM location is accurate to 1 meter.

Suppose the target decryption region starts from location $P_0(x_0, y_0)$. The decryption region is assumed to be a square region. A square region must have two components: a starting point and length (length equals to width). The location of the decryption vehicle is known. Since the decryption vehicle will be placed in the center of the decryption region (to tolerate location errors), the starting location of the decryption region can be computed if we know the decryption region size. Therefore, only the length of the square needs to be determined. Based on UTM location accuracy, the length of the square L is listed as follows 10, 100, and 1000 meters. For 10-digit UTM locations, $1 < L < 10^4$ because the accuracy is about 1 meter. For 8-digit UTM locations, $10 < L < 10^7$ because the location is accurate to 10 meters. For 6-digit UTM locations, $L < 10^4$ because 6 digits UTM locations are accurate to 100 meters. No UTM location smaller than 6 digits can be used in our algorithm. Therefore, the length of the square is selected from one of the three possible lengths: 10, 100, and 1000 meters.

The decryption region can be computed on the basis of the map of roads and the

mobility of vehicles. The methods that we address are the following:

1. The location of communication peers can be calculated on the basis of the mobility parameters including current speed, location, acceleration, etc. This is a dynamic computation method. A new location after a certain time interval can be computed. Suppose at time t_0, the target vehicle is at location (x_0, y_0) with speed v_{x0}, v_{y0} and acceleration a_{x0}, a_{y0}, where x_0, v_{x0}, a_{x0} are the x-axis value of initial location, relative speed on x-axis direction, and relative acceleration on x-axis direction; y_0, v_{y0}, a_{y0} are the y-axis value of initial location, relative speed on y-axis direction, and relative acceleration on y-axis direction. After time interval t, we can roughly predict that the vehicle will be at a place near location region x_1, or

$$x_1 \in [x_0 + v_0 t + \frac{1}{2} a_0 t^2 - \alpha * XDeviation, \\ x_0 + v_0 t + \frac{1}{2} a_0 t^2 + \alpha * XDeviation] \tag{62}$$

and y_1, or

$$y_1 \in [y_0 + v_0 t + \frac{1}{2} a_0 t^2 - \alpha * YDeviation, \\ x_0 + v_0 t + \frac{1}{2} a_0 t^2 + \alpha * YDeviation] \tag{63}$$

where x_1 and $XDeviation$ are the x-axis location prediction and the x-axis location deviation values, y_1 and $YDeviation$ are the y-axis location prediction and the y-axis location deviation values, and α is the coefficient which implies the effect of the deviation, $0 \geq \alpha \geq 1$.

2. If the decryption region is a fixed area, we can directly check the map of roads and calculate the GPS coordinates. This is the simplest scenario.

3. If the decryption region is dynamically moving, we can calculate the location of the decryption region by querying the target receiver.

Updating the Decryption Region

Although the decryption region can be predicted, there may be prediction errors because of the high mobility of vehicles. Therefore, the decryption region needs to

be corrected to improve prediction accuracy for future communication. The predicted location will be updated by the real location which is piggybacked on other communication messages. The speed, acceleration and direction of movement will be piggybacked as well. Therefore, the updating step includes the following assignments:

$$x_1 = x_{real} \tag{64}$$

$$y_1 = y_{real} \tag{65}$$

$$XDeviation = (1 - \beta) * XDeviation + \beta * |x_{real} - x_0| \tag{66}$$

$$YDeviation = (1 - \beta) * YDeviation + \beta * |y_{real} - y_0| \tag{67}$$

where (x_{real}, y_{real}) is the real location piggybacked, and β is the coefficient value which implies the effect of the prediction error $|x_{real} - x_0|$.

The updating frequency depends on the mobility of receiving vehicles, the accuracy requirement of the decryption region and the bandwidth of the control channel. For example, the frequency of updating on highways should be much higher than the frequency of updating in urban areas because the velocities on the highway are much higher than the ones in urban areas. Similarly, the accuracy of the decryption region and the bandwidth of the control channel impact the updating frequency as well. The more frequently the updating occurs, the more accurate is the location that can be obtained. On the other hand, the more frequently the updating occurs, the higher the overhead of control messages and the more bandwidth will be wasted. There is an optimal value of the frequency of updating message. We can obtain this value by simulating and engineering methods which we discuss in Chapter VI.3.

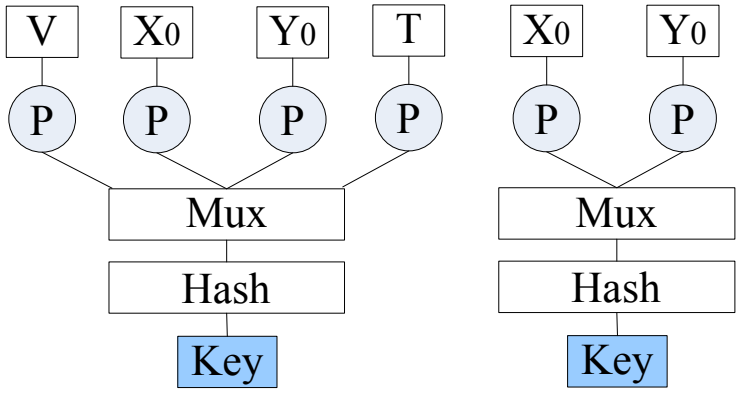
V.2.2 D-GeoLock Mapping Function

The main function of GeoLock is to convert a location into a secret key which will be used in encryption and decryption. The GeoLock mapping function in Denning's GeoEncryption algorithm is a lookup table which has been shown in Figure 45. The major drawbacks include 1) expensive deployment to all vehicles, 2) difficult to update on all vehicles, and 3) low location error tolerance. Therefore, we propose D-GeoLock algorithm which can overcome these drawbacks. Our contribution of the mapping function is to convert a secret key value on the fly. There are no preinstalled mapping tables in our algorithm. Our D-GeoLock mapping algorithm converts VANET parameters into a unique value as a secret key. There are several

parameters: location coordinates (x_0, y_0), time interval T, and speed interval V.

From the Sender's Perspective

The process of generating a lock value/key is shown in Figure 49. The input parameters are speed V, location coordinates (x_0, y_0), and time T for the comprehensive D-GeoLock shown in 49(a). The input parameters are location coordinates (x_0, y_0) for the location D-GeoLock shown in 49(b). The output result is a unique string/number. There are three steps. Step one will process the input parameters. The location (x_0, y_0) will be divided by the length of decryption region (square) L. For example, if the length of target decryption region is 100 meters or $L = 100$, then each of coordinate numbers of $P_0(x_0, y_0)$ will be divided by 100. Only the integral part after division will be obtained. Therefore, larger values of L will result in fewer digital numbers of the output from step one. Fewer digital numbers will result in a weaker lock key. If the value of L is small, there is a risk that a lock key may be computed by a brute force attack. Step two will multiplex the outcome of previous processing step and form a unique string, i.e. the output of the first step is multiplexed or reshuffled. Step three will compute the hash value of the outcome from step two. The reason for step three is to equalize the size of the keys. The hash function in practice can be designed as mod operation or as standard hash function, such as Secure Hash Algorithm (SHA).



(a) Comprehensive D-GeoLock. (b) Location D-GeoLock.

FIG. 49: D-GeoLock mapping function.

An example of D-GeoLock is shown in Figure 50. We show only location D-GeoLock as a simplification for the example. The comprehensive D-GeoLock is the

future work. In the first step, two numbers are divided by the length of the region 100, i.e. (042.00, 915.00). The integer part after division, i.e. (042, 915) is kept. In the second step, the two numbers: (042,915) are multiplexed or concatenated as 042915. In the third step, the multiplexed number is hashed by SHA to generate the lock value.

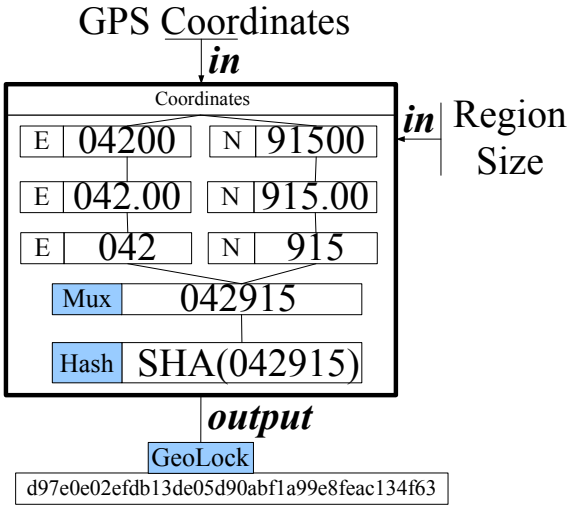


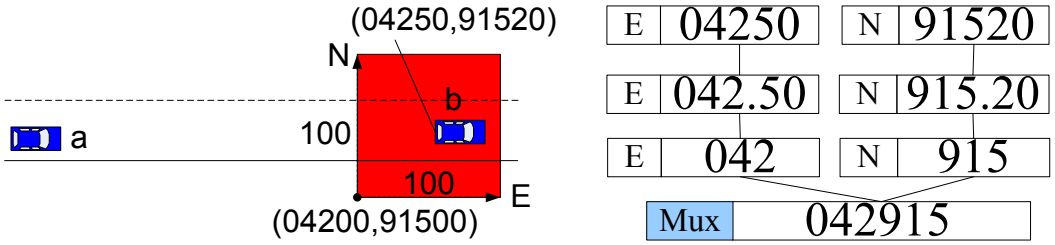
FIG. 50: An example of D-GeoLock.

V.2.3 From the Recipient's Perspective

The receiver needs to convert its own location into the secret key using the D-GeoLock algorithm. Figure 51 shows how the receiver can convert its location into the secret key. The receiver can obtain V, X_0, Y_0 and T from the enlisted GPS receiver. The same D-GeoLock mapping function discussed in Section V.2.2 is used to compute a lock key. If vehicle b is restricted by the decryption region in terms of location, the exact same lock value will be generated. Otherwise, the lock key will be different. Thus, the secret key Key_S will not be recovered and the ciphertext $E\{Req\}$ will not be decrypted.

An example of the mapping function on the receiver's view is shown as Figures 51.a and 51.b. The receiver vehicle b is located at location $(04250, 91520)$ (UTM 10 digital coordinates) shown in Figure 51.a and the decryption region L is 100 meters. Figure 51.b shows the D-GeoLock process on the recipient. Although the receiver's location coordinates are $(04250, 91520)$, the new D-GeoLock algorithm generates exactly same key as the sender generates. It is obvious that the vehicles will pass the

geographic validation.



(a) A decryption region specified by a location and a length.

(b) Computing decryption key.

FIG. 51: An example of secret key recovery.

V.3 SUMMARY

In this chapter, we have addressed a geographic location based encryption, D-GeoEncryption. D-GeoEncryption will specify a decryption region where vehicles must be physically present to decrypt the message. This requirement can enhance confidentiality.

The GeoEncryption algorithm started from the introduction of the original algorithm proposed by Denning [79]. Denning's GeoEncryption algorithm uses preinstalled tables to convert location into a secret key. Given the large scale of vehicle population, it is very hard or expensive to install and update the preinstalled tables on all vehicles. Denning's GeoEncryption algorithm also uses a single session key. We improve upon this by dynamically changing multiple session keys. To improve GeoLock, we specially design a new GeoLock which can dynamically compute the secret keys from location inputs. In our new GeoLock algorithm, there are no preinstalled tables.

CHAPTER VI

EVALUATION

In this chapter we evaluate our three proposed location security mechanisms, location availability, location integrity and location confidentiality. For location availability, we first evaluate the link duration model and link duration probability expressions. We validate the relationship between the link duration and the mobility parameters (speed and acceleration) in several cases. The pdf of the link duration is also checked. Routing paths on the basis of link duration model are also simulated. We are interested in the duration of routing paths, the overhead of the control packets, and the throughput of the routing path. For location integrity, we validate location information and identify the malicious attackers who send out fabricated location claims. We are interested in detection time, detection ratio, and abnormal location identification. For location confidentiality, we compare our D-GeoEncryption algorithm with another extension of the original GeoEncryption algorithm. We are interested in the decryption ratio and the cost of updating the decryption region.

VI.1 AVAILABILITY SIMULATION

Vehicle mobility, including speed, acceleration and direction, dynamically changes the topology of vehicles. The communication links between vehicles, therefore, are unstable, and location availability is inherently affected. In our solution, we improved location availability by proposing a reliable routing scheme based on our link duration model and on probability of link duration. The purpose of the simulations in this section is to validate our link duration model and the probability expressions of link duration. We also evaluate the performance of our location availability technique in terms of the duration of the routing path, the overhead of control messages and the throughput of the routing path.

Based on IEEE 802.11p and Dedicated Short Range Communications (DSRC), vehicles with wireless devices can communicate with other vehicles and with roadside infrastructure, if any exists. Since the FCC has restricted DSRC communication to around 300m, multiple hops are needed to send messages to their destinations. Besides, the topology of the network changes frequently due to rapid vehicular mobility. The mobility model adopted in this work is the Intelligent Intelligent Driver

Model (IDM) [98, 115]. Here, vehicles can accelerate/decelerate, change lanes and stop when blocked.

The nodes in VANETs have high mobility. Therefore, simulations of VANETs typically require a networking simulator and a mobility simulator. In our simulations, the mobility simulator simulates the traffic dynamics and produces a trace file of vehicle movements. The trace file records the movement of vehicles. The general format of the movement is: <ID> <time> <speed> <x-coordinate> <y-coordinate> <z-coordinate>. The network simulator imports the trace file and moves nodes based on the movement track provided. The network simulator simulates network performance in the presence of the mobility of nodes. Our mobility simulations were performed under the mobility simulation platform of Treiber [98]. We recorded the trace files of vehicle mobility and then imported them into NS-2 (2.30). In NS-2, each node represents one vehicle in the mobility simulations, moving based on the represented vehicle movement history. Nodes in NS-2 are also entities of wireless communication nodes integrating network protocol stacks.

VI.1.1 Probability Model Simulation

The general parameters we used in simulations are detailed in Table 2. The total number of vehicles n is varying because in our mobility simulation, vehicles may enter and exit. The number of vehicles on the road at any time depends on the headway distance. We use the Constant Bit Rate (CBR) application in NS-2 to generate network traffic. The network protocol is 802.11. The total number of network connections is one third of the total number of vehicles n. Some connections break after a certain time. At that point, we create new connections to replace the broken connections. We use a straight highway to simulate the road. The road length is 10 miles, and the traffic density is 1500 vehicles per hour. The general link duration model parameters are shown in Table 3.

There are three cases in the simulations. Specifically,

Case I Both vehicle i and j have positive speeds and accelerations. They are in the same direction as well. Case I covers the scenarios originally shown in Figure 20.e-f and shown here in Figure 52.

TABLE 2: Availability Simulation: Environment Configure

Name	Value
Total number of vehicles (n)	1000-2000
Application	CBR (128Kbps)
Network protocol	IEEE 802.11
Network connections	$\lfloor n/3 \rfloor$
Simulation map	Highway
Road length	10 miles
Total number of lanes	2
Traffic density	1500 vehicles/hour

TABLE 3: Availability Simulation: General Parameters

μ	The log-normal distribution parameter	2.95
σ	The log-normal distribution parameter	0.55
f	The frequency of wireless channel	5.9 GHz
γ	The power fall-off coefficient in (5)	4
σ_g	The power coefficient for R_g in (5)	7.0
C_1	The coefficient in (11)	0.5
X	The initial link distance	100m
v_m	The velocity	33m/s
n	The number of vehicles	1500

Case II Both vehicle i and j have positive speeds and the same moving direction but they have opposite accelerations (for example, i has a positive acceleration and j has a negative acceleration). Case II covers the scenarios originally shown in Figure 20.a-d and shown here in Figure 53.

Case III Both vehicle i and j have opposite speeds, opposite accelerations and opposite directions. Case III covers the scenarios originally shown in Figure 21.e-f and shown here in Figure 54.

Case I and II are in the co-directional, as shown in Figure 55. Case III is in the opposite direction, as shown in Figure 56. The rest of scenarios, e.g. Figure 21.a-d and Figure 22, are not covered because they are not frequently occurring scenarios.

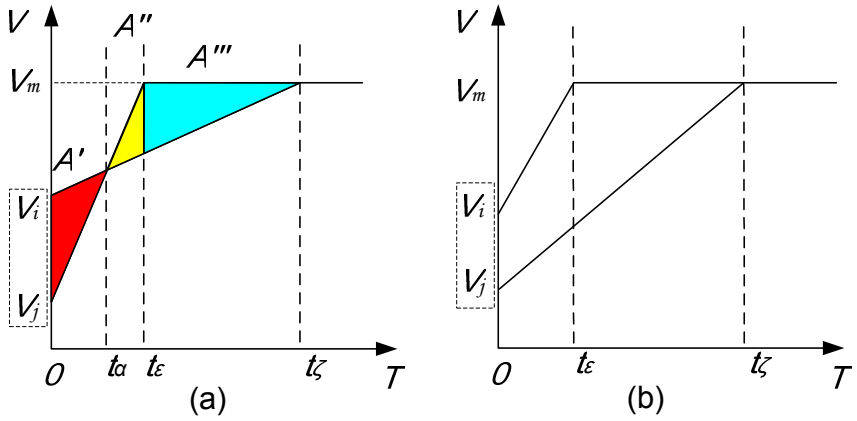


FIG. 52: Simulation scenario Case I. Co-directional vehicles with positive speed and positive acceleration.

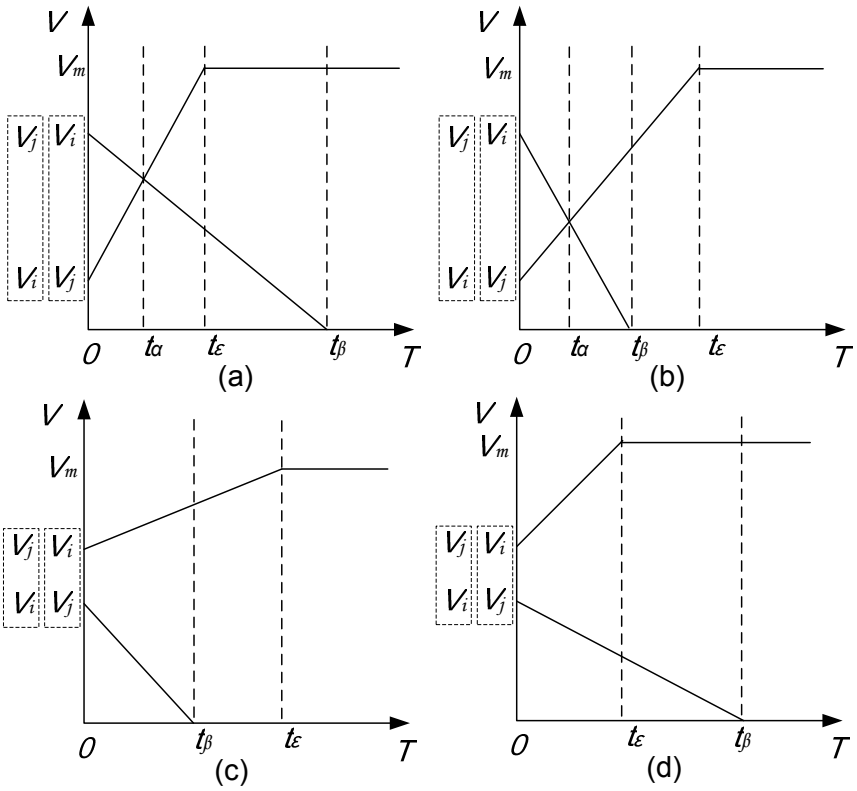


FIG. 53: Simulation scenario Case II. Co-directional vehicles with positive speed and oppositive acceleration.

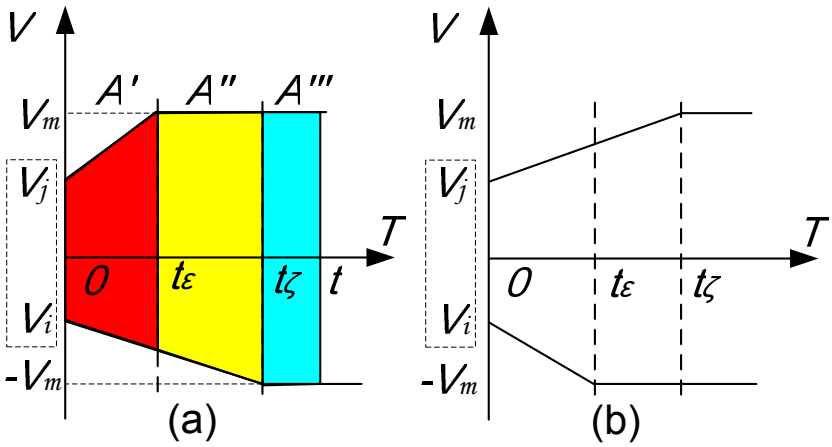


FIG. 54: Simulation scenario Case III. Opposite-directional vehicles with opposite speeds and opposite accelerations.

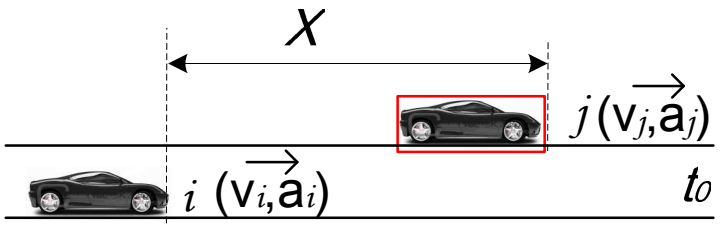


FIG. 55: Simulation Cases I and II. Both vehicles are in the same direction.

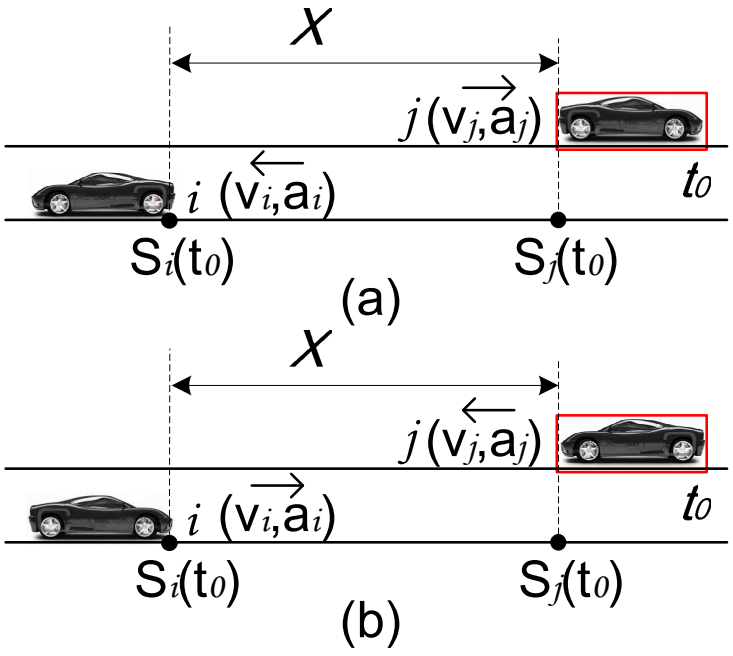


FIG. 56: Simulation Case III. The two vehicles travel in the opposite direction.

Velocity Simulation

To begin, we were interested in the relationship between the speed and link duration in the three simulation cases. The purpose of this simulation is to verify our theoretical expression of link duration, given speed and acceleration of both i and j. The parameters we used are shown in Table 4. The parameters μ and σ are from the log-normal distribution of headway distance. Given $\sigma = 0.55$, we assume that the mean value of headway distance is 30 meters, i.e. $e^{\mu+\sigma^2/2} = 30$ from which μ can be determined. f is the frequency of wireless channel. In USA, f is 5.9 GHz for DSRC. The values of γ and σ_g are for the log-distance path loss model [116] and [117]. C_1 and C_2 are the normalizing coefficients in (11), i.e.

$$F(z) = \frac{C_1 a}{2}\left[1 + \text{erf}\left(\frac{z-\mu_1}{\sigma_1\sqrt{2}}\right)\right].$$

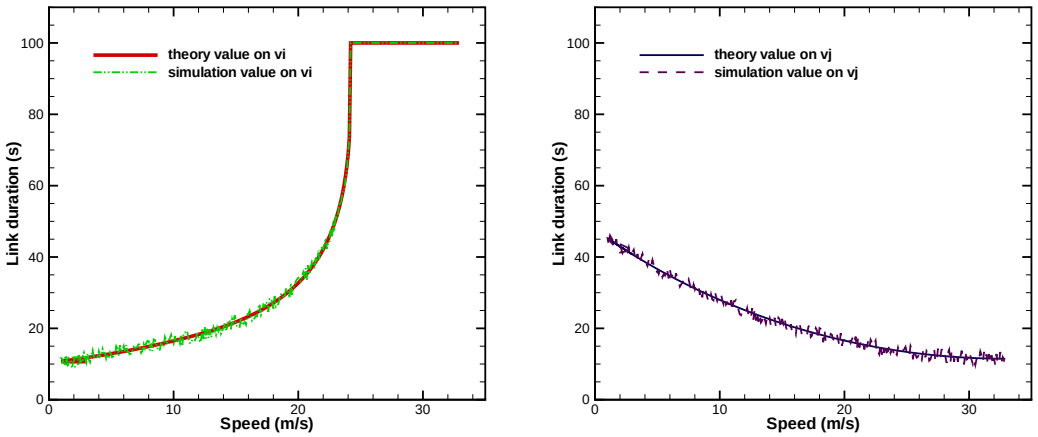
The simulation settings of Cases I, II and III, in this section, are to validate the relationship between the speed and the link duration.

TABLE 4: Velocity Simulation: Mobility Parameters

	Case I	Case II	Case III
a_i (The acceleration of i) (m/s^2)	0.1	-1	-1
a_j (The acceleration of j) (m/s^2)	1	1	1
v_{i0} (The initial velocity of i) (m/s)	15	20	-31
v_{j0} (The initial velocity of j) (m/s)	15	10	30

We varied the speed (both v_i and v_j) from 0.5m/s to 33m/s in Cases I, II, and III and recorded the link duration. The link duration of 100 seconds represents infinite link duration. The initial speed and acceleration values are shown in Table 4. When we changed the initial speed of i (v_{i0}), we kept v_{j0} unchanged. If we changed the initial speed of j (v_{j0}), we kept v_{i0} unchanged. Each step of speed change is 0.1m/s.

Figure 57 shows the impact of speed on link duration for Case I. A higher speed for vehicle i will cause a longer link duration, and a higher speed for vehicle j will cause a shorter link duration, because both i and j are accelerating in the same direction as shown in Figure 55. When the initial speed v_{i0} increases, it increases the likelihood of reaching v_m. Therefore both i and i reach v_m and they are moving parallelly. The link tends to be stable and can last forever (i.e. more than 100s).



(a) Speed v_i. (b) Speed v_j.

FIG. 57: Speed impact in Case I.

The results of Case II are shown in Figure 58. As we expected, the simulation results match with the theoretical values. The higher speed of vehicle j will cause the shorter link duration, and the higher speed of vehicle i will cause a longer link duration. We can explain the result by recalling the link duration model. If vehicle j has faster speed, the relative speed between j and i, i.e. $v_j - v_i$ will be higher. Therefore, the link duration will be shorter. We know the distance covered by i at time t is

$$S_i(t) = \int_0^t v_i(x)dx.$$

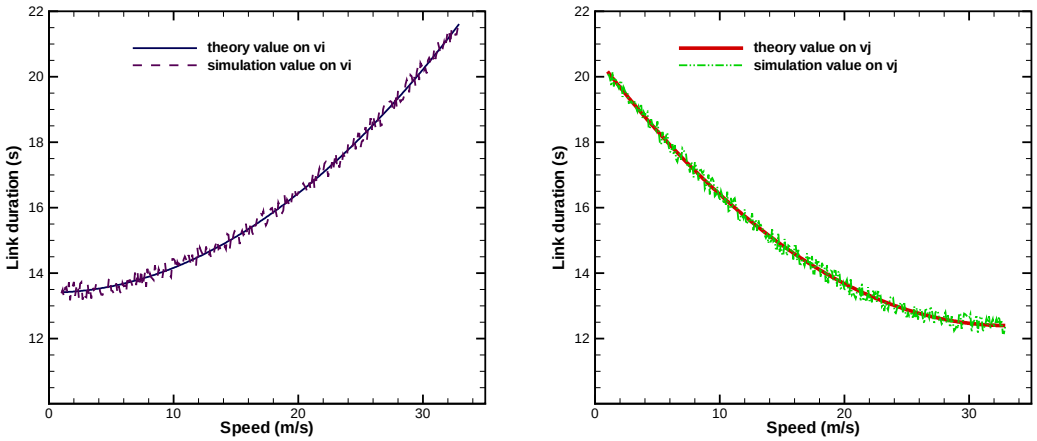
This is the area covered by function $v_i(t)$ in Figure 59. The distance covered by i at time t is

$$S_j(t) = \int_0^t v_j(x)dx.$$

This is the area covered by function $v_j(t)$ in Figure 59. Therefore, the distance between the sender and the receiver is

$$S_j(t) - S_i(t).$$

As shown in Figure 59, the shaded areas represent the distance between the two vehicles. In the simulation, we obtained one set of link duration values by changing the value of $v_i(0)$ and keeping the value of $v_j(0)$ unchanged. Since the link break distance is 300m, the link will break if the two cars move apart by 200m because the



(a) Speed v_i.

(b) Speed v_j.

FIG. 58: Speed impact in Case II.

initial link distance was 100m. If $v_i(0)$ is larger, the line of v_i will parallelly move up and the shaded area with 200 will move right. Therefore, the link duration will be longer. We obtained another set of link duration values by changing the value of $v_j(0)$ and keeping the value of $v_i(0)$ unchanged. If $v_j(0)$ is higher, the line of v_j will parallelly move up and the shaded area with 200 will move left. Therefore, the link duration will be smaller.

The simulation setting for Case III is included to validate the relationship between the speed and the link duration in formula A.2.1.b (in the Appendix). The results of Case III are shown in Figure 60. As we expected, the simulation results match the theoretical values. The higher speed of vehicle i will cause a longer link duration, and the higher speed of vehicle j will cause a shorter link duration. In Case III, both i and j are accelerating in opposite direction. We can explain the result using the Case III scenario shown in Figure 61. When $v_j(0)$ increases, the line of v_j moves up, and the area of 200m will move left. Therefore, the link duration will decrease. When $v_i(0)$ increases, the line of v_i moves up because v_i is negative, and the area of 200m will move right. Therefore, the link duration will increase.

Acceleration Simulation

The previous simulation showed how varying speed affects link duration. Since acceleration affects link duration as well, we simulated relationship between the link

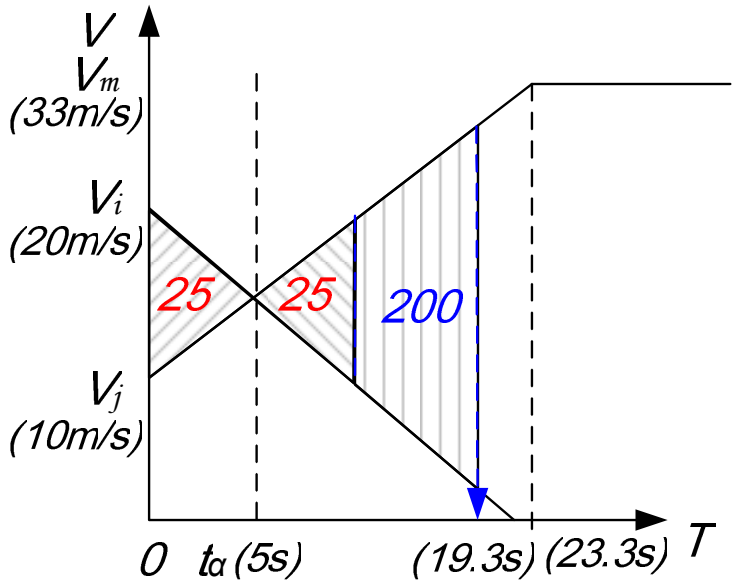
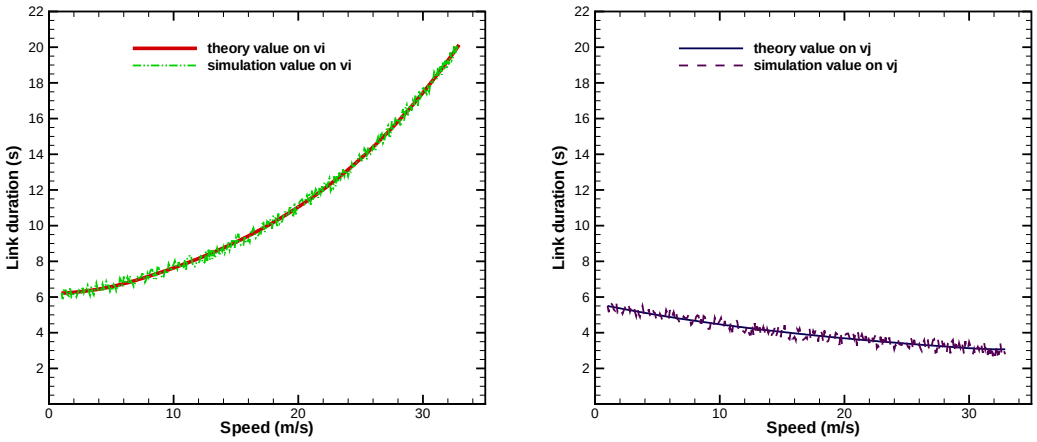


FIG. 59: An example scenario for simulation Case II. When $v_i(0) = 20m/s$, $v_j(0) = 10m/s$, $a_i = -1m/s^2$ and $a_j = 1m/s^2$, the link duration is 19.3s. The numbers in the shaded areas are the distance values between the sender and the receiver.



(a) Speed v_i.

(b) Speed v_j.

FIG. 60: Speed impact in Case III.

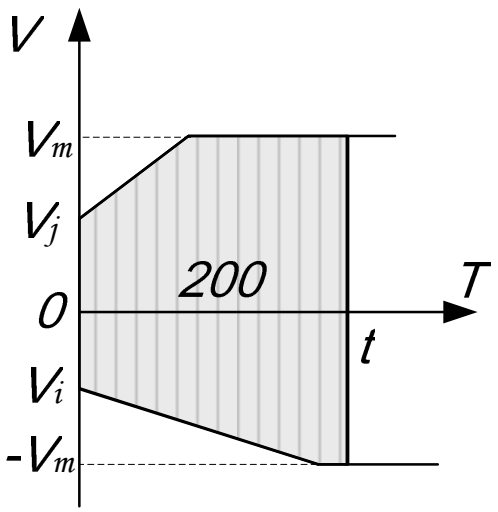


FIG. 61: The scenario for the results of Case III.

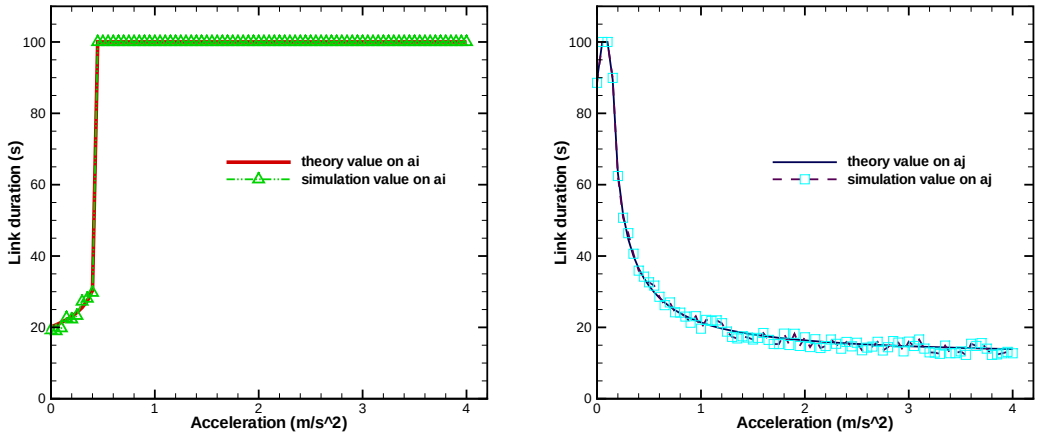
duration and acceleration in Case I, II, and III. We only modified the mobility parameters compared with the previous simulations. The initial speed and acceleration values are changed because we want to simulate longer link durations. We increased the relative speed in Case I and decreased the relative speed in Case III. The change of the mobility parameters is shown in Table 5. Simulation results are generated by varying acceleration and recording the link duration.

TABLE 5: Acceleration Simulation: Mobility Parameters

	Case I	Case II	Case IIII
a_i (The acceleration of i) (m/s^2)	0.2	-1	-1
a_j (The acceleration of j) (m/s^2)	1	1	1
v_{i0} (The initial velocity of i) (m/s)	30	20	-5
v_{j0} (The initial velocity of j) (m/s)	1	10	3

The results of Case I are shown in Figure 62. As we expected, the simulation results match with the theoretical values. The increment of a_j will cause the decrement of link duration, and the increment of a_i will cause the increment of link duration. The increment of a_i will cause the speed v_i to quickly reach the speed limit. Therefor, both vehicles are moving at the same speed and the link duration tends to be forever (more than 100s). On the other hand, if the acceleration a_j increases, the speed v_j will quickly reach a high speed and the relative speed will increase. The link is more

likely broken. The acceleration a_i is more sensitive than a_j because it is more likely that i and j will be in a state where both vehicles are traveling at the limit speed v_m.



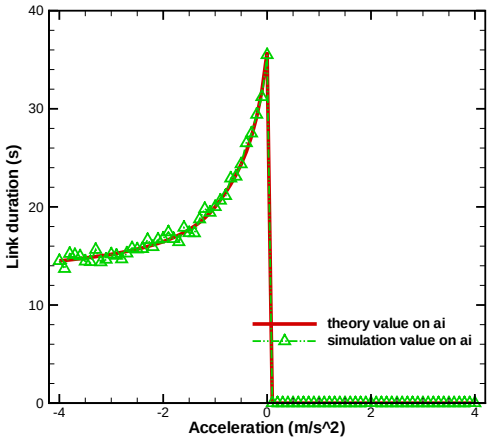
(a) Acceleration a_i. (b) Acceleration a_j.

FIG. 62: Acceleration impact in Case I.

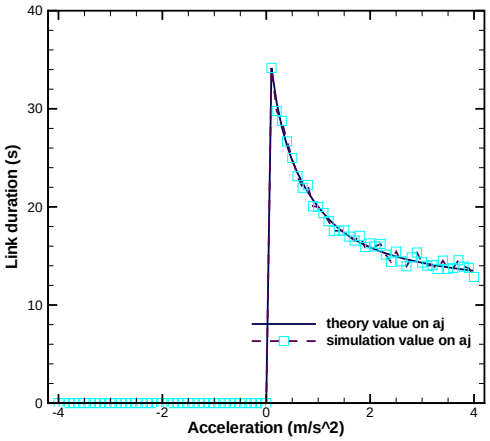
The results of Case II are shown in Figure 63. As we expected, the simulation results match with the theoretical values. The increment of a_i will cause the increment of link duration, and the increment of a_j will cause the decrement of link duration. The smaller acceleration of i means that i stops quickly because the acceleration of i (i.e. a_i) is negative. The relative speed between i and j will increase, and the link duration will decrease. On the other hand, the faster acceleration of j means quicker speed increment of v_j. The relative speed will be larger, and the link duration will be smaller.

The results of Case III are shown in Figure 64. As we expected, the simulation results match with the theoretical values. The increment of a_i will cause the decrement of link duration, and the increment of a_j will cause the increment of link duration. In Case III, the acceleration of i is negative. The smaller a_i will cause the faster increment of i in the opposite direction of j. Therefore, the relative speed between i and j will be larger and the link duration will be smaller. On the other hand, the larger a_j will result in the faster increment of j. Therefore, the relative speed between i and j will be larger and the link duration will be smaller.

We were also interested in the pdf of link duration. Since our theory shows that the link duration is a form of the log-normal distribution, the purpose of this

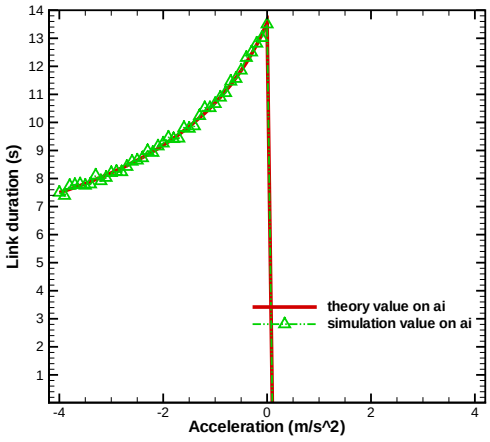


(a) Acceleration a_i.

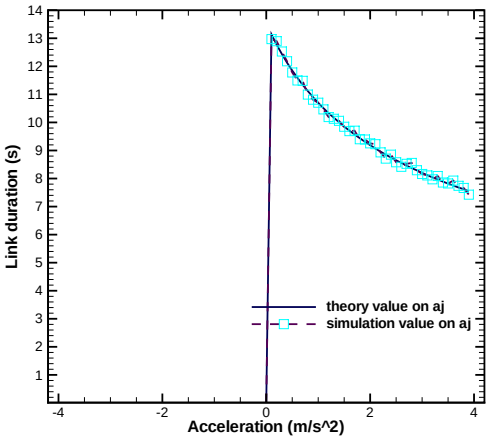


(b) Acceleration a_j.

FIG. 63: Acceleration impact in Case II.



(a) Acceleration a_i.



(b) Acceleration a_j.

FIG. 64: Acceleration impact in Case III.

simulation is to validate the pdf of link duration which is covered in Chapter III.1.3. For each case of Case I, II, and III, we created 2000 links among vehicles and recorded the link durations of each. As we expected, the pdf of link duration for each case is a form of log-normal distribution, shown in Figure 65. Case I has a flat curve, but Case II and III have relatively sharp curves. This is because Case II and III tend to form a centralized relative speed and acceleration. Therefore the link duration values are crowded around a certain range.

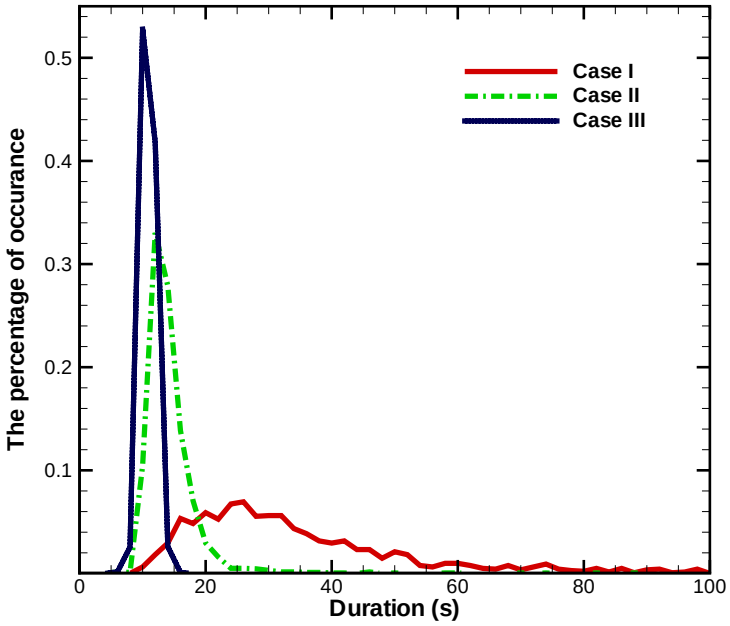


FIG. 65: The pdf of link duration.

VI.1.2 Routing Protocol Simulation

In this section, we used the same simulation environment as the one used in Section VI.1.1. The simulation map is a straight highway which is 5 Km long. The initial speed is 25 m/s and the average speed is 28 m/s. The acceleration is in the range from 0 to $2m/s^2$. The detailed parameters used in this section are shown in Table 6.

We simulated our routing protocol and compared it with DSR (Dynamic Source Routing) [33] and ROMSGP (Receive On Most Stable Group-Path) [118]. DSR is a well-known on-demand routing protocol. DSR records the source-destination path information during route discovery. The learned source-destination path information

TABLE 6: Routing Protocol Simulation: Environment Configure

Name	Value
The number of vehicles (n)	1000-2000
Application	CBR
Network protocol	IEEE 802.11
Transmission range	300m
Network connections	$n/3$
Simulation map	Highway
Road length	5 Km
Traffic density	1500 vehicles/hour
Average speed	28 m/s
Acceleration range	[0,2] m/s^2
Initial acceleration	0 m/s^2
Initial speed	25 m/s
Mobility model	IDM

is used to route packets. DSR was originally proposed for MANETs, but DSR has been refined and applied in VANETs [52]. ROMSGP is a mobility-based routing protocol for VANETs. ROMSGP computes the link duration time of a group of vehicles. By selecting the most stable routing link, ROMSGP can route packets for a group using a stable routing path. The difference between ROMSGP and our proposed routing scheme is that our scheme computes not only link duration but also the probability of the link duration. In addition, our computation is based on the mobility parameters speed and acceleration, but ROMSGP's computation is based only on speed.

First, we investigated the average duration of the routing path. The source and destination nodes are randomly selected among the vehicles on the highway. For each routing path, we computed and recorded the average path duration and the average speed of the vehicles involved in the routing path. As expected, our protocol provides more stable paths than the other two protocols. This is because we select the most durable routing links, as shown in Figure 66. Our protocol is designed on the basis of link duration and the probability of the link duration. For each link, we selected the optimal one, and thus the optimal routing path can be obtained. We also dynamically maintain the routing path by locally replacing and globally reconstructing the routing path.

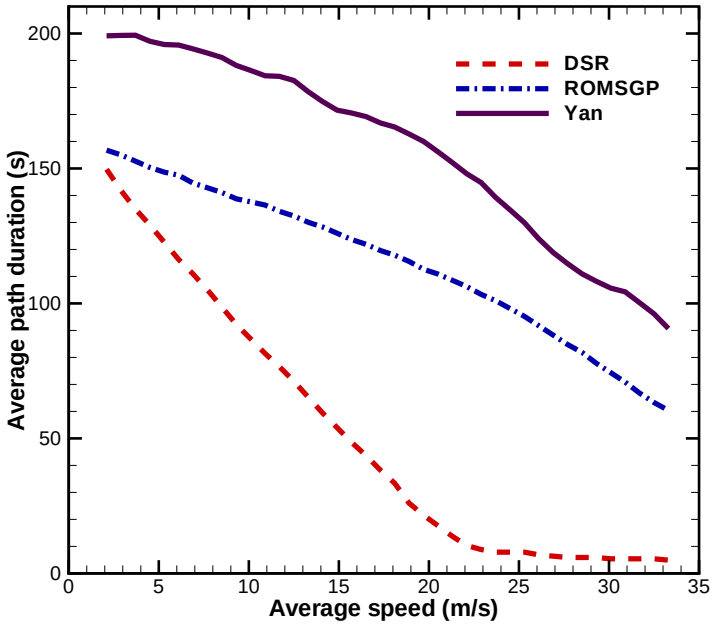


FIG. 66: Duration of links.

We were also interested in the control packet overhead. Vehicles in VANETs will periodically change mobility status to maintain an optimal routing path and to keep the routing path alive as long as possible. Therefore, there will be some control packets exchanged among vehicles. The control packets consume network resources and are treated as overhead. In this simulation, we counted the number of control packets among vehicles involved in the routing path and the average speed of the involved vehicles. We plotted the number of control packets and the average speed in Figure 67. As expected, our proposal is more efficient than DSR and ROMSGP. This is because our maintenance of the routing path is lightweight. If the expected duration of the routing path is t, we maintain the routing path at time $t/3$ and $2t/3$. Therefore, we only send out a few packets. DSR and ROMSGP are constantly sending control packets to maintain the routing path.

For many applications, throughput is one of the key factors. We compared our protocol with the other two protocols in terms of throughput. In this simulation, we computed the average throughput of the routing path and the average speed of the involved vehicles. We plotted the average throughput and the average speed in Figure 68. As expected, our proposed protocol still outperforms both DSR and ROMSGP since fewer routing links will break under our protocol. This is because

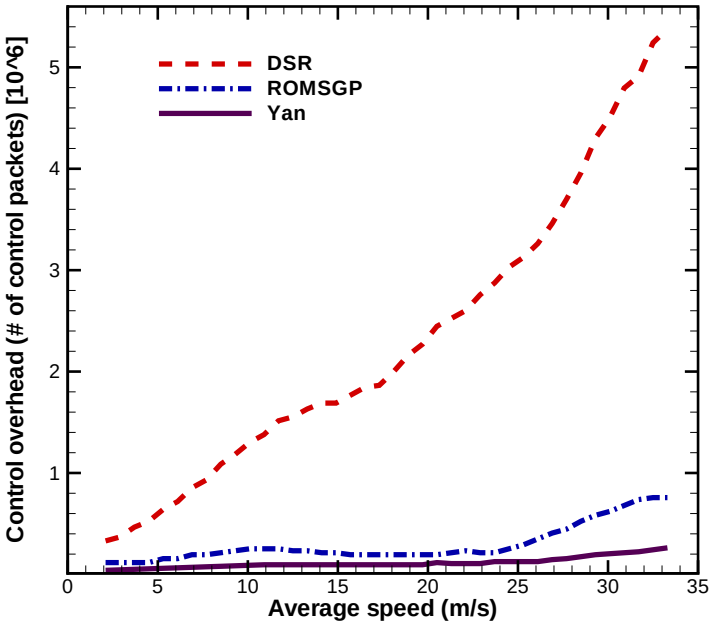


FIG. 67: Control message overhead.

our protocol is more reliable and stable. There are fewer breaks of the routing path, therefore, vehicles in the routing path can continually exchange packets. We also noticed that the throughput decreases when the speed of vehicles increases. This is because more links will break when the speed is high.

VI.2 INTEGRITY SIMULATION

The purpose of location integrity is to validate location information. We proposed three solutions: active location integrity, passive location integrity, and general location integrity. The active location integrity solution validates the received location by using installed radar. The passive location integrity solution filters and refines the location information about an observed vehicle. The general location integrity combines the active and passive location integrity solutions. Therefore, to evaluate integrity, we introduce some malicious vehicles that send fabricated packets in the simulation. We use a self-developed Java simulator to identify the malicious vehicles. We are interested in the detection time, the detection ratio, and the abnormal location identification.

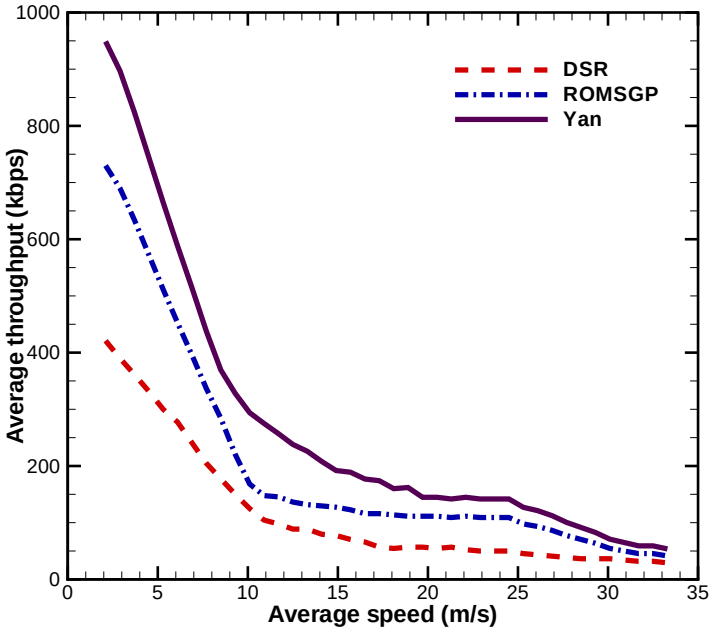


FIG. 68: Throughput.

VI.2.1 Validating Locations

In our experiments, we simulated a bidirectional 3 km highway with two lanes in each direction. The cell radius was 100 meters, traffic arrival rate was 1600 vehicles/hour, mean velocity was 33.3 m/s, and transmission radius initially was 100 meters. Each simulation has some number of compromised, or malicious, vehicles and a single observer vehicle. When the observer enters the simulated highway, it initiates a request to find all of the compromised vehicles. The simulation terminates when the observer reaches the end of the 3 km highway.

We wanted to investigate the amount of time required to detect a certain number of compromised vehicles. In our first set of experiments, we inserted 16 compromised vehicles and varied the total number of vehicles on the highway. Each experiment was run 10 times, and we measured the average amount of time needed to detect the 16 compromised vehicles. As a comparison, we show in Figure 69 the average amount of time to detect the compromised vehicles using our cell-based routing as opposed to message flooding, where each node re-broadcasts the received message. As expected, cell-based routing is more efficient than flooding. The time needed to detect the compromised vehicles increases when the number of vehicles decreases.

The reason is that in low density traffic vehicles must physically drive the packet to the next cell if there are no middle nodes available as routing nodes.

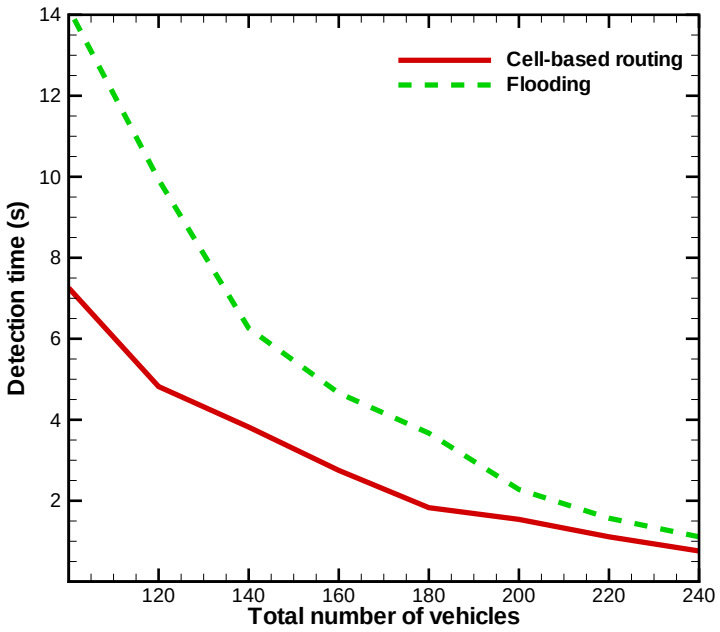


FIG. 69: Time needed to detect the 16 malicious vehicles as the total number of vehicles varies.

To determine how transmission range affects the time needed to detect malicious vehicles, we ran a set of experiments with a 100 m transmission range and a set with a 500 m transmission range. The vehicle density was about 30 vehicles per kilometer per lane on the highway. The compromised vehicles were 5% of the total vehicles and were randomly deployed along the highway. In each set of simulations, we varied the length of the highway to investigate the effect of transmission range. Since the time depends on the number of intermediate hops, the increased range of transmission would certainly decrease the time. However, the increased transmission range increases the probability of packet collisions. Figure 70 shows the time taken to detect the malicious vehicles with respect to their distance from the vehicle generating the verification request. As expected, with a 500 m range, the time is much lower, but the re-broadcasting of packets needs to be handled carefully to avoid an increase in collisions.

Finally, we wanted to investigate how many compromised vehicles could remain undetected in our system. We randomly distributed compromised vehicles along the highway. The vehicle density was about 30 vehicles per km on the highway. The

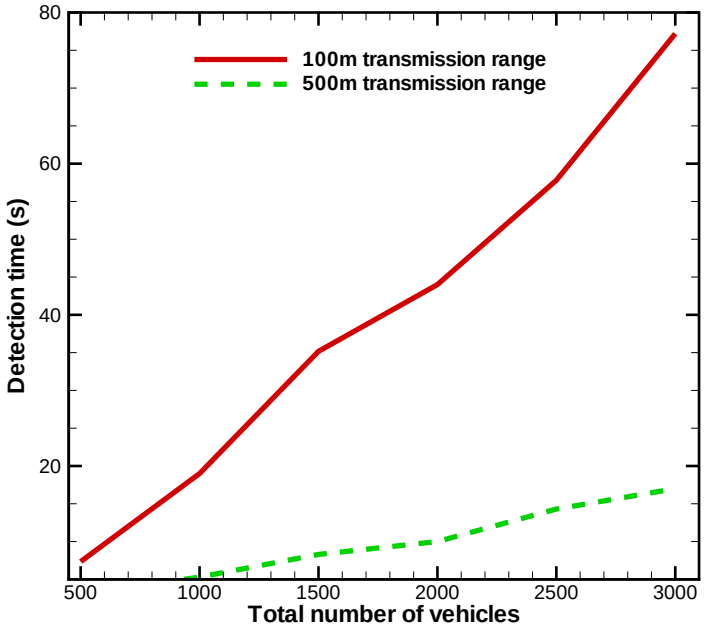


FIG. 70: Average detection time.

transmission radius was 100 meters. We varied the percentage of all vehicles that were compromised from 0% to 30% because our assumption is that the majority of vehicles are honest. Compared with previous compromised vehicle percentage (5%), we increased this number to 30% to investigate a larger scope. We measured the number of compromised vehicles detected in 30 seconds and show the results in Figure 71. As expected, the percentage undetected compromised vehicles decreases when the percentage of compromised vehicles decreases.

VI.2.2 Filtering and Refining Location

In the simulation of filtering and refining location solutions, we assumed 1% of vehicles will send out compromised locations. They are malicious attackers who can create a random bogus location. In our simulations, the malicious vehicles always send false locations. The remaining 99% vehicles are honest regarding the location information. All vehicles report what they detect or forward what they receive. The observed locations have measurement errors which are normally distributed with mean μ and deviation σ. Two vehicles are monitored. One is the target vehicle X which is observed by its neighboring vehicles. Another is the observer vehicle which

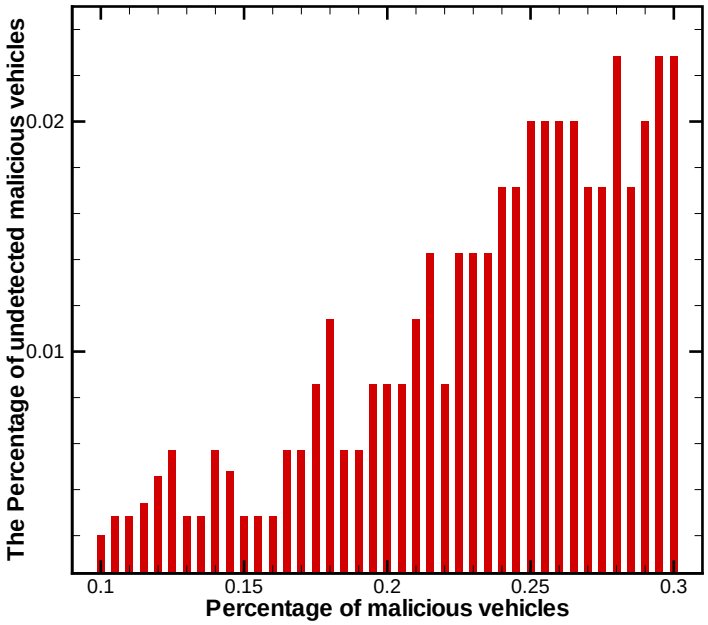


FIG. 71: The number of undetected compromised vehicles as the percentage of compromised vehicles increases.

collects the location of X. The vehicle X announces its location and sends it to the observer vehicle. X's neighboring vehicles, receiving this announcement in 300 meters, will figure out the location of X using the received signal strength indicator [119, 120]. The neighboring vehicles will send their measurements. The malicious attackers will send their bogus location as well. The observer vehicle collects all of these reports and treats them as input. The input was processed by MATLAB R2009A where our algorithms were implemented. The initial simulation parameters and values are listed in Table 7. The initial traffic density is 35 vehicles per Km per lane. The road is a straight highway which is 3Km long. The average speed of vehicles is 60 Km/h. The number of lanes is 4 per direction. The mean error and the deviation of the reported locations are both 1m. The acceptable location estimation error is 3m. The number of outliers is 4 for the neighboring vehicles and 1 for opposite vehicles. The weights of final location estimation in (59) are 0.5 for w_1, 0.3 for w_2 and 0.2 for w_3.

In our simulation results, a Q-Q plot (Quantile-Quantile Plot) [121] is applied to show the Mahalanobis distance vs. the normal quantile. The Q-Q plot is a graphical method for comparing two probability distributions. It is a commonly used tool in

TABLE 7: General Integrity Case: Parameters and Values

Parameters	Values
Initial traffic density	30 vehicles/Km/lane
The length of the road L	3 Km
Average speed	60 km/h
The number of lanes	4/direction
The mean error μ	1 m
The deviation of error σ	1 m
Error ϵ	3 m
# of neighbor outliers m_n	4
# of opposite outliers m_o	1
The weight for radar w_1	0.5
The weight for opposite w_2	0.3
The weight for neighbors w_3	0.2

statistics to show outliers. Quantiles are points taken at regular intervals from the cumulative distribution function (CDF) of a random variable.

We were interested to show the Mahalanobis distance of reported locations from neighboring vehicles and opposite vehicles. The samples from radar are not included because samples from radar are detected by the vehicle itself which will not contaminate the location detection. In the simulation, each vehicle will report one location of the observed vehicle to the observer vehicle. The observer vehicle will collect 990 location reports from normal vehicles and 10 location malicious reports (outliers) from attackers. Figure 72 shows the Mahalanobis distance for neighboring samples. The Mahalanobis distance computed from outliers are far away from the normal samples. As expected, the Mahalanobis distance computed from malicious reports can be identified and expelled from the normal data.

We were also interested in the final location estimation to validate the proposed method in this work. On the basis of the previous simulation, we can obtain the filtered location observations. These observations are resampled and manipulated to obtain the location estimation by using bootstraping method discussed in Section IV.3.3. Figure 73 shows the plot of the location observations and the location estimation. We obtained the location estimation coordinates: $(-0.0041642037, -0.002087865)$. Compared to the real location $(0, 0)$, the precisions of x-y axis are about $(0.416\%, 0.2\%)$.

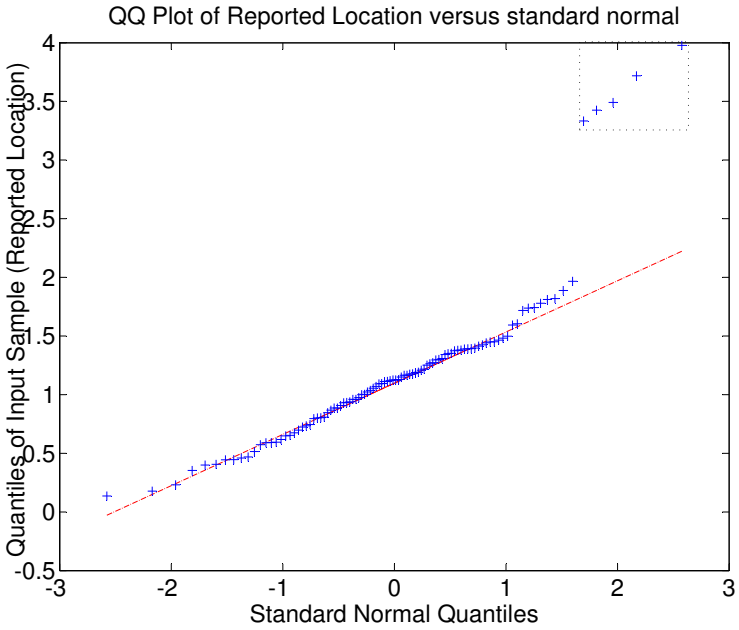


FIG. 72: Q-Q plot of the Mahalanobis distance for neighboring samples. Each point represents one Mahalanobis distance computed from one sample. The rectangle indicates all the outliers which are away from the other samples. The normal samples are aligned to the straight line.

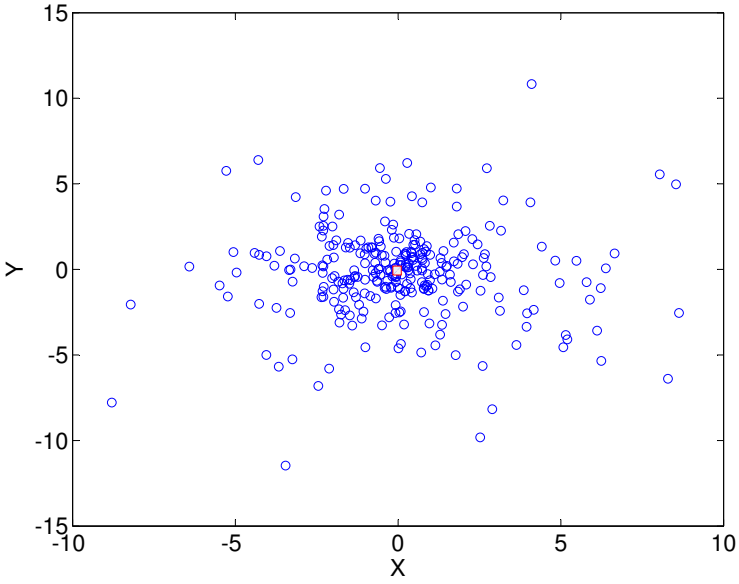


FIG. 73: The x-y coordinates of location observation and the location estimation. Each point represents one location observation. The rectangle represents location estimation.

VI.3 CONFIDENTIALITY SIMULATION

In this simulation, we simulated our D-GeoEncryption algorithm to enhance the location confidentiality. Instead of using a straight highway, we simulated an urban scenario which is generated by the SUMO simulator [122]. SUMO is an open source and microscopic road traffic simulation package which is often used by transportation researchers. The network simulator we used is ns-2 [123]. SUMO creates a trace file which records the mobility of vehicles. We loaded the trace file into ns-2 to evaluate the security techniques. The application used is Constant Bit Rate (CBR) with 128Kbps. The total number of vehicles is 320. The map is 3.2km x 3.2km. The number of encryption-decryption peers is 20. The decryption region of vehicles is a square of 100m x 100m. The simulation settings are shown in Table 8.

TABLE 8: The selected environment configuration

Name	Value
Transmission range	300m
Simulation map	Urban
Map area	$3.2 \times 3.2\ Km^2$
Decryption area	$100 \times 100\ m^2$
Traffic density	1500 vehicles/hour
Average speed	28 m/s
Acceleration range	[0,2] m/s^2
Initial acceleration	0 m/s^2
Initial speed	25 m/s
Mobility model	IDM [Treiber *et al.* (2000)]

Since our interest is in location-based security, we predicted a decryption region based on the updated vehicle's location. The vehicle's decryption region is dynamically updated on the basis of the vehicle's mobility. We recorded two events:

- Decryption failure, a message is not able to be decrypted.
- Decryption success, a message is successfully decrypted.

We compared our algorithm with Al-Fuqaha's algorithm. Al-Fuqaha *et al.* [80] extended the Denning's GeoEncyption by predicting and updating the decryption region along with mobile nodes. Al-Fuqaha's algorithm still uses the table-based GeoLock to convert a location to a secret key.

In the GeoEncryption algorithm, a location is converted to a secret key for encryption and decryption. We are interested in the decryption ratio which is defined as the number of successfully decrypted messages over the number of the received messages. We varied the location error from 0 to 10%. We compared two sets of results with a speed 24 m/s. The result of the comparison is shown in Figure 74. As we expected, our proposed algorithm has a much higher decryption ratio than Al-Fuqaha's algorithm. This is because of the design of the D-GeoLock. In our al-

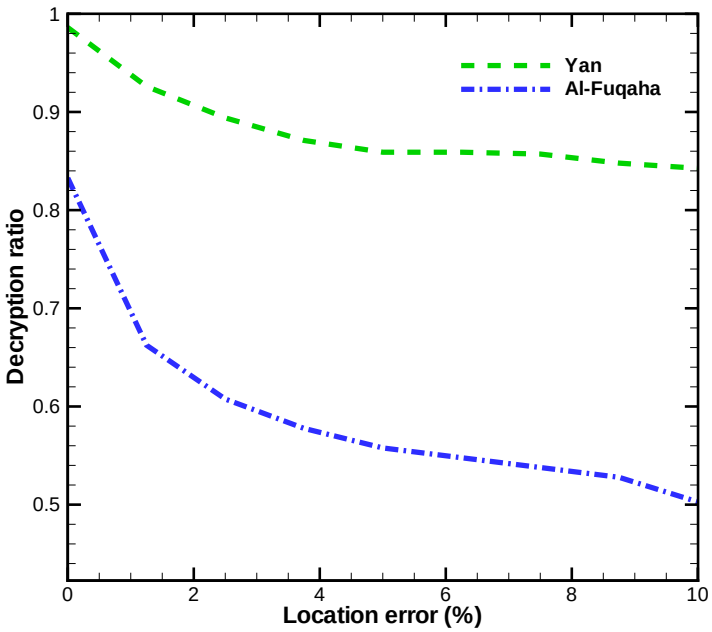


FIG. 74: Decryption ratio.

gorithm, the location is divided by the size of the decryption square and the integer part is collected. Our algorithm can tolerate location errors. For example, a location (04200, 91500) can produce a multiplexed outcome of 042915. The location with measurement error (04299, 91599) can produce the same multiplexed outcome, as shown in Figure 75. Even if two locations can end up with the same keys, the private key of the receiver is needed to decrypt a message.

Relative mobility is one of the major challenges in the D-GeoEncryption algorithm. We wanted to know how much relative speed can affect the decryption ratio. We recorded the communication of encryption-decryption peers and the speed of each peer. We computed the relative speed by using the average speed of each peer.

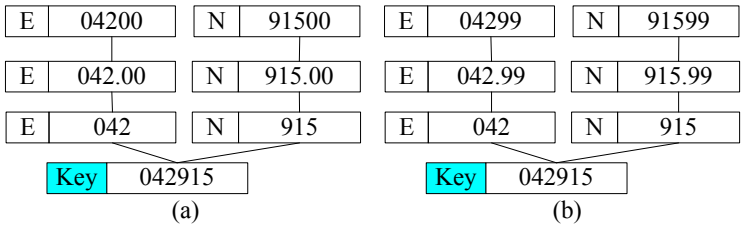


FIG. 75: Our GeoLock can tolerate location errors. Both cases with different locations obtain the same multiplexed outcome.

As expected, the decryption ratio of our algorithm is less affected by the relative speed, as shown in Figure 76. We also noticed that the decryption ratio decreases when the relative speed increases. This is because the interval of updating location information between the encryption-decryption peers is $\triangle t$ which is a fixed value. The location change in the interval $L = \triangle t * v_r$ where v_r is the relative speed. Larger relative speeds will cause larger location changes, which will cause smaller decryption ratios.

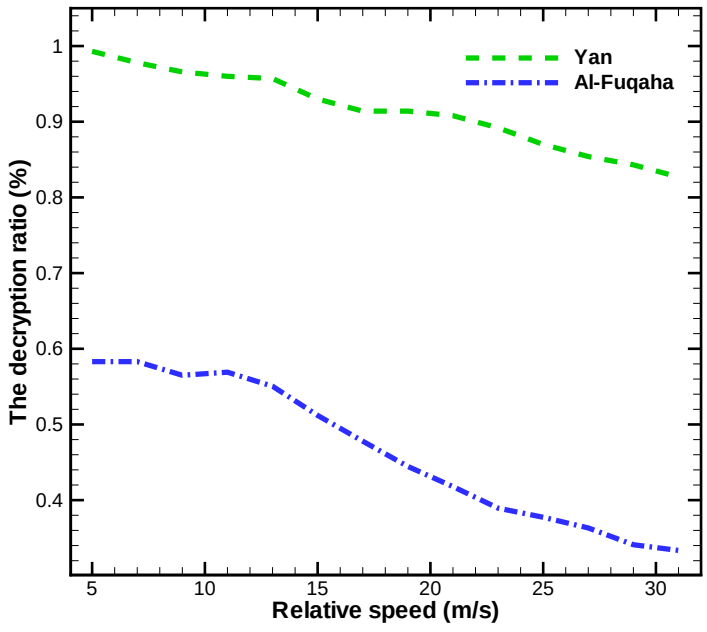


FIG. 76: Relative speed can affect the decryption ratio

Security is not free but has a cost. In the GeoEncyption algorithm, the decryption

region will be predicted and updated. Vehicles have to exchange locations by sending control messages. We investigated the number of control messages as overhead. The control messages can be sent with a certain frequency. We varied the updating intervals which are the inverse of the updating frequencies. The control message ratio and the decryption error ratio are counted and computed for each interval. The control message ratio is defined as the number of the control messages divided by the total messages. The decryption error ratio is defined as the number of failed decryptions divided by the number of received messages. As expected and as shown in Figure 77, our algorithm outperformed Al-Fuqaha's algorithm. The increment of the updating intervals results in a lower updating frequency. The lower updating frequencies will cause larger location errors which will result in larger decryption errors. On the other hand, the lower updating frequencies can lead to a smaller number of control messages. Interestingly, we can find a joint point between the curve of the overhead results and the curve of the decryption error results, as shown in Figure 77. The joint point shows the optimal updating interval which can lead to a compromise for both the overhead and the decryption errors. Our algorithm has an optimal updating interval of 1.6s while Al-Fuqaha's algorithm has the value of 3.6s. Therefore, our algorithm can send fewer control message to achieve the same decryption ratio as Al-Fuqaha's algorithm.

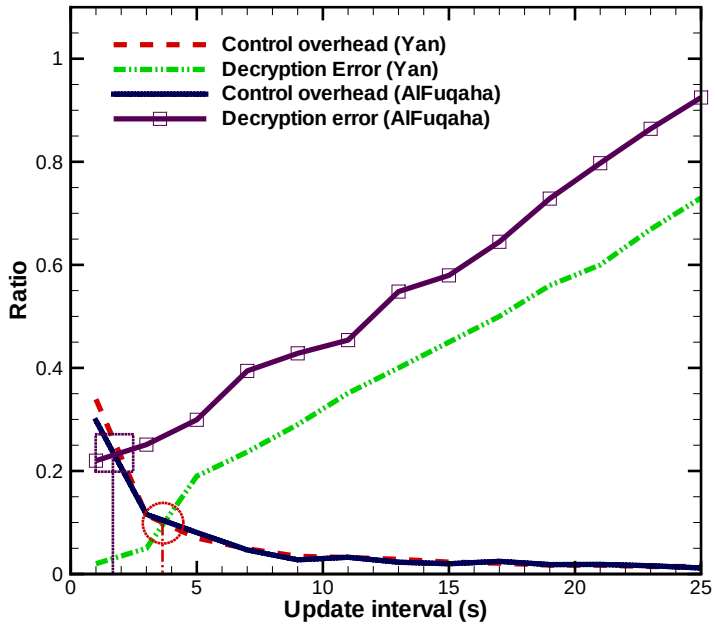


FIG. 77: Overhead of D-GeoEncryption.

VI.4 SUMMARY

In this section, we explained in detail our simulation results. Our simulations include location availability, location integrity and location confidentiality simulations. For location availability, we were interested in the pdf of link duration, the duration of links, and the impact of speed and acceleration to the link duration. We also examined the throughput and control overhead. Simulations showed that our location availability is not only effective but also efficient. For location integrity, we were interested in the detection time of attackers and the filtering of attackers. Simulations showed the location integrity mechanism can effectively validate the location modification and detect attackers. In location confidentiality, we were interested in the decryption ratio and the overhead of our algorithm. Simulations showed that our D-GeoEncryption has high decryption ratio, high location error tolerance, and low overhead.

CHAPTER VII

SUMMARY AND FUTURE WORK

In this chapter we summarize the motivation for this dissertation, the problems we have addressed, and the solutions we have proposed. The work in this dissertation also projects future research. We also list some future developments and extensions to our work. The chapter is organized as follows: Section VII.1 addresses the summary and main conclusions of this dissertation. Section VII.2 analyzes the proposed security techniques that can prevent most location attacks. The simulation results are listed in Section VII.3. The contributions of this dissertation are listed in Section VII.4. Future extensions and developments of this work are shown in Section VII.5.

VII.1 SUMMARY

Vehicles installed with advanced devices can communicate with each other and create wireless networks, called VANETs. The applications developed for this network will bring a revolution to our driving experience. The applications of VANET include safety and entertainment applications. Most, if not all, of these applications are location-aware and strongly depend on location information. Compared with other networks, such as MANETs, the Internet, and cellular networks, VANETs have unique and distinctive features:

- large scale of node mobility,
- large scale of node population,
- physical limitation of node mobility by roadways,
- short wireless communication range.

In this dissertation, we are protecting a tuple, correct time, correct identity, correct location, although we simply call it location information. We initially show that location security is of fundamental importance to VANET applications. Based on information security requirements, we have to ensure location information confidentiality, integrity and availability. The following attacks can harm location security:

- Dropping messages will break the routing path. This kind of attack will block *location availability*. Without location availability, you cannot get location information when you need it most.
- Modifying the content of location information. This damages *location integrity*.
- Replaying information. This is an attack that sends outdated messages to pretend that the previously past vehicles still exist. This attack violates *location integrity*.
- Injecting location messages with fabricated locations and IDs. A Sybil attack is an attack that send bogus identities with location information to create an illusion that traffic is jammed. This attack damages *location integrity*.
- Eavesdropping on messages and then decrypting it to access location information. This kind of attack will expose location information to people who have not been authorized. This breaks *location confidentiality*.

Previous research can only protect part of the three requirements. For example, PKI is used to encrypt VANET information such as locations to protect location confidentiality. The digital signatures of messages are used to protect location integrity. But both PKI-based and digital-signature-based methods cannot improve location availability which is the fundamental requirement of location security. In addition, PKI encryption and digital signature processing usually come with expensive processing time, for example PKI encryption time is three times more computationally expensive than symmetric encryption e.g. DES3. Other security methods such as signal-based methods [71, 72] and resource-based methods [73, 20] can only validate location integrity without providing location confidentiality and availability. The methods in the literature have a common drawback in that none of them can provide an ultimate and comprehensive location security solution.

Therefore, the research in this dissertation addresses location information security by providing location information confidentiality, integrity, and availability. The presentation of the research is organized into location confidentiality, location integrity, and location availability on the basis of the information security *CIA* model.

Location information needs to be propagated to other vehicles for both safety and comfort applications. Due to the high mobility of vehicles, the wireless connections among vehicles are not stable. Location information routing is inherently unreliable.

Therefore location availability will be harmed. We improve *location availability* by proposing a reliable routing scheme which is on the basis of the link duration model. In the link duration model, we assume that the inter-car headway distance (i.e., the instantaneous gap between consecutive vehicles) obeys a log-normal distribution. Using this assumption and a classic radio propagation model, we derive the probability and the mean duration of a link.

A reliable routing path is selected on the basis of the link duration model. The source vehicle broadcasts a probing request *Prb* on the wireless channel. All vehicles in the transmission range receive the request and compute the distance from the sender, the duration of the link, and the probability of the existence of the link. An acknowledgment packet *AckP* will be constructed that contains the computed distance, duration, and probability of the link. The *AckP* is sent back to the sender who will collect multiple *AckP*s and select the best link *i.e.*, the one with the highest probability and whose expected duration matches the routing path duration requirement *EDur*. The sender then sends a confirmation packet *CfmP* to the next hop on the best link. This node will further explore the routing path by broadcasting the *Prb*. When the destination node receives the *Prb*, it will terminate the probing process and will send an *AckP* back to the source vehicle along the newly formed routing path. This completes the routing path exploration stage.

The basic idea of maintaining a routing path is to repair those links that are expected to break soon. The expiring links can be predicted using the link duration estimation. These links can be locally repaired by replacing them with new links formed by neighboring vehicles or globally reconstructed using the path discovery protocol discussed above. For a local repair, the destination vehicle initiates the process by sending a routing break packet *RBRK* at the expected expiration time of the routing path (*i.e.*, the minimum duration of the links). Vehicles which comprise the weakest link replace it with more reliable links. A global repair is similar to finding an entirely new routing path.

In our *location integrity* model, we began by presenting our first contribution, an active location integrity mechanism. This is motivated by the need to provide secure topology information in VANET and to build a secure network for applications. Underlying our solution is the well-known adage: "*Seeing is believing*". We assume vehicles are equipped with GPS, an on-board radar device, a computer and a wireless transceiver. In our scheme, the on-board radar acts as a virtual eye of the vehicle,

while the wireless transceiver acts as a virtual ear of the vehicle. Although the eyesight is rather limited due to the modest radar transmission range, a vehicle can see the surrounding vehicles and hear reports of their GPS coordinates. We expect the on-board radar device to provide useful corroboration of alleged location information, except for during short transient periods. For example, the line-of-sight that radar needs may be temporarily obstructed by a large truck. Due to the dynamic nature of traffic, even if there are transient obstructions, the line of sight will eventually be restored.

To overcome the inherent range limitations of radar and the wireless transceiver, we build network cells as basic security and communication units. The cells are sized such that each vehicle can directly communicate with all other vehicles in its own cell. To achieve intra-cell security, a vehicle may use its radar to verify the alleged position of neighboring vehicles in its own cell. Thus, it is reasonable to assume that each vehicle knows, with high certainty, the position of the other vehicles in its own cell. At the same time, since many applications require position information beyond the current cell boundary, we propose a method for inter-cell position information integrity. In our scheme, when a vehicle receives an aggregated message which alleges the position of a remote vehicle, it can randomly challenge the position of a vehicle in a remote cell. The challenging vehicle uses the on-board radar of remote vehicles or enlists the help of vehicles in the oncoming traffic. When radar is not available, vehicles can rely on reports from oncoming traffic or trusted neighbors.

We propose a method to validate position information in scenarios where an on-board radar device is not available or its use is hampered by various traffic/weather conditions. Specifically, we propose a passive location integrity solution. We forgo the requirement of an on-board radar device but assume the presence of on-board GPS and transceiver. Vehicles collect location information from neighbors and oncoming vehicles. After validating and filtering out false or inaccurate locations using a data fusion algorithm, vehicles store the validated locations into memory to create a track-record (called a Map History) of the mobility of other vehicles. Based on the Map History, position prediction can be used to validate the announced positions. The data fusion algorithm achieves intra-cell location integrity by counting the number of location reports and abandoning reports that highly deviate from the majority of reports. For inter-cell location integrity, we apply the intra-cell position validation method in the cell that includes the announced position.

As VANET technology will be deployed incrementally, not all vehicles will have on-board radar, GPS, and a transceiver. Some vehicles will have all three, some will have only GPS and a transceiver, while others will have none of the devices. Mindful of this, we propose a validation system that works in this scenario. Vehicles will create a Map History table using radar-validated locations, oncoming traffic detected positions, and neighbor location information to achieve location integrity. Each type of data is given a weight, and we apply data fusion algorithms to filter out spurious location information.

To ensure inter-cell location integrity, the aggregated position information of several vehicles is transmitted over the wireless medium which is open to the public. If the aggregated message is in plaintext, it is vulnerable to an assortment of attacks. One simple solution is to encrypt the plaintext message by using conventional cryptography (*e.g.*, using symmetric or asymmetric keys). However, key management is a very challenging task given the huge number of participating vehicles. In addition, attackers can crack conventional cryptography by employing several well-known techniques. To provide *location confidentiality*, we propose a geographic location-based security mechanism to provide physical security on top of conventional methods. Messages are encrypted with a geographic location key that specifies a decryption region. This provides physical security because a vehicle has to be physically present in the decryption region in order to decrypt ciphertext encrypted with this geographic location key. This physical security needs to encrypt messages with a secret key which is converted from geographic location information. We design a new conversion algorithm that can compute the secret key from the geographic location information.

VII.2 SECURITY ANALYSIS

The methods proposed in this dissertation can prevent most position attacks in VANETs. We now show how each type of attack is addressed by our scheme:

- Fabrication attacks: the location integrity module has three schemes to filter out fabricated location information (active, passive and general location integrity).
- Alteration attacks: the location confidentiality module can prevent location information from being modified by unauthorized vehicles.

- Packet dropping: the location availability module can overcome malicious packet dropping and establish a reliable routing path that avoids intentional packet dropping.

- Replaying: the location integrity module can validate the correctness of location information at the specified time and vehicle ID.

VII.3 SIMULATION RESULTS

There are two mobility simulators that we used in our simulations, SUMO [122] for urban scenarios and a tailored Java simulation engine [98] for highways. Our network simulator is NS-2.30 [123]. The mobility simulators generate a trace file which records the mobility of each vehicle. This trace file is imported into NS-2 to control the movement of each vehicle. We summarize our findings in the following points:

1. Relative velocity plays a vital role in link duration. Positive relative velocity will dramatically decrease the link duration. It is possible to reach infinite link duration if two vehicles are traveling in the same direction and with a consistent zero relative speed.

2. Relative acceleration also plays an important role in link duration. Positive relative acceleration will dramatically decrease the link duration.

3. Our link duration model and link probability model are verified by simulation. The formula of link duration is based on the mobility of two vehicles. The relationship of the mobility parameters has been examined and verified. The pdf of link duration is proven to be a log-normal distribution.

4. Our location availability routing protocol, developed on the basis of our link duration model, has more than 80% longer path durations than DSR and more than 40% long path durations than ROMSGP. The overhead of our method is only 1/3 of ROMSGP and 1/50 of DSR. The throughput of our method is more than 25% higher than ROMSGP and about 50% higher than DSR.

5. For location integrity, the cell-based network is more efficient than the flooding network. The cell-based network needs about 50% less time to detect attacks.

6. The data fusion mechanism proposed in this dissertation performed correctly and filtered 100% of wrong location information.

7. Our D-GeoLock algorithm can tolerate larger location errors than the original GeoLock algorithm. The geographic location-based encryption stated in our dissertation has about 70% higher decryption ratio than AlFuqaha's method.

8. Our D-Geoencryption algorithm has lower control overhead than the original Geoencryption algorithm.

VII.4 CONTRIBUTIONS

Our motivation in this dissertation was to provide location security in terms of location confidentiality, integrity and availability. The main contribution is the first comprehensive solution to enhance location security. We provide location confidentiality, integrity and availability. Specifically, in location availability, we discover that the link duration distribution of VANETs has a log-normal distribution which was often assumed in MANETs to be exponential. On the basis of this discovery, we proposed a reliable routing protocol to improve location availability. Our protocol can reduce control overhead and reduce response time. In location confidentiality, a dynamic GeoLock (D-GeoLock) algorithm is proposed. D-GeoLock can dynamically convert a geographic location into a secret key without checking location mapping tables. Compared to the original GeoLock, D-GeoLock can tolerate larger location errors. On the basis of D-GeoLock, D-GeoEncryption algorithm is addressed, which can specify a decryption region where messages can be decrypted. In location integrity, we are the first to propose active location integrity using on-board radar, and we address a real world location integrity solution.

VII.5 FUTURE WORK

Preliminary work on the location information security was completed as part of this research. The need for efficient implementation has never been greater than now. As part of our future work, we plan to continue to improve the current implementation, develop new methods and conduct scalability studies under different traffic situations. We present possible future work in this section.

VII.5.1 Location Availability

In our current work, we analyzed vehicular mobility to obtain a link duration model and a link probability model. This model works on a highway scenario because most of the vehicles are traveling in the same direction and are constrained by the roadway. Although our proposed link duration model and link probability model can work in an urban/city scenario, we would have to frequently update the mobility information of vehicles so that they could be aware of the status of neighbors. This will cause certain overhead of control messages. In the future work of location availability, we can include digital maps to know the road type such as "T" type of intersections, "Y" type of intersections, "+" type of intersections, etc. Once we know the road type, we can know when we need to collect the status of neighboring vehicles.

VII.5.2 Location Integrity

We introduced radar as the "virtual eye" of vehicles because some new models of vehicles have radar installed for cruise control. Actually, there are some other devices that have been installed in cars, such as camera, infrared detector, etc. These devices can serve as eye devices of cars as well. Local location integrity can be achieved using these devices. For example, a car-mounted camera can be used to detect and predict vehicle accidents. The algorithm for predicting accidents includes location prediction as well. We can incorporate the location prediction algorithm for cameras into our location integrity in future work.

For passive location integrity, we can use different statistical methods to filter the incorrect location information and refine low-resolution location information to high-resolution location information. For example, we can apply Bayesian Probability to compute the confidence probability of location information on the basis of certain events such as changing lanes. With the probability of location information, we can fine-tune location detection.

The scenarios of sparse traffic can change the communication of VANETs. Figure 78 shows gaps between clusters of vehicles when the traffic is sparse. The election of the cell routers and leaders will take a long time. In this type of scenario, the cell routers and the cell leaders are not applicable for VANETs. Cells discussed in IV are not applicable any more as well. The different communication styles require different location integrity solutions. New solutions that can work in sparse traffic

are the future focus work to improve location integrity.

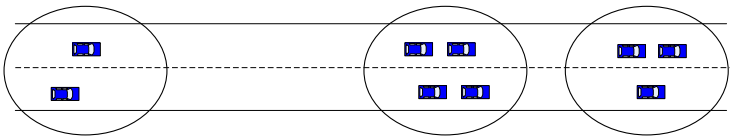


FIG. 78: Gaps created when the traffic is sparse.

VII.5.3 Location Confidentiality

As future work, we can integrate the D-GeoEncryption method into existing security methods. Different applications will need different levels of security. For example, an expensive-cargo-tracking application may need high security of location. A medical assistant application may need high security as well. PKI-based location confidentiality can be applied. Many roadside location eavesdropping attacks may be defeated by a simple XOR encryption. Safety-related applications, such as traffic condition alert, forward/rear collision warning, and pedestrian crossing alert, do not need any encryption. Therefore, we can implement different levels of location confidentiality for different applications.

In our current design of location confidentiality, the decryption region is square-shaped. As future work, the shape of the decryption region can be generalized to any shape. One idea to do this is to partition an irregular shape into a set of regular shapes. For each obtained regular shape, we can apply the GeoEncryption algorithm. Then we can merge the subtasks of location confidentiality to obtain the whole location confidentiality.

Vehicles in traffic often form clusters or groups as illustrated in Figure 78. The D-GeoEncryption algorithm can be extended in this scenario. A group of vehicles can share a secret key to encrypt and decrypt messages. The advantage of this idea includes:

- providing efficient key management. One group key can serve all the members in a group.
- reducing control message overhead. The groups will be the communication unit. The number of players is reduced and fewer control messages are sent.

Another future work topic is the new algorithm of GeoLock. For example, time and velocity can be included in the new GeoLock algorithm. In the battle field application, we can provide more strict location security. If vehicles are moving at or above a certain speed and before a certain time, they are allowed to decrypt the received messages. Otherwise, they are not allowed to decrypt the messages.

VII.5.4 Miscellaneous Issues

We now briefly list other possible future work:

- Cross-layer research on location information security. For example, we can detect the location of vehicles by computing the wireless signal strength at the physical layer, conduct location data fusion at the network layer, and we can encrypt location messages based on geo-encryption algorithm at the application layer.

- Infrastructure-based location security. By using infrastructure, we can obtain accurate and reliable location information about the whole topology of network.

- Extensively simulate the proposed methods to evaluate the performance. For example, we can collect real traffic data using cameras and retrieve location information to obtain location information. The obtained location information can serve as our input of mobility of vehicles. We can simulate our ideas on the basis of real traffic.

- Conduct scalability research in different traffic situations by real or close-to-real traffic data to study the impact on performance from different schemes.

- Integrate with other research such as privacy protection. In this dissertation, we only discuss location security without touching privacy protection. But privacy protection is as important as location security because location is one piece of the privacy information of passengers. The basic idea of privacy protection is pseudonymization which will give vehicles a temporary pseudonym with a certain expiration time.

APPENDIX A

PROBABILITY MODEL APPENDIX

A.1 FITTING THE PDF OF HEADWAY

To fit the simulation result and pdf curve by lognormal distribution, we define a *general form* of the lognormal distribution.

$$f(x,\mu,\sigma,a,b,c) = \frac{b}{(cx-a)\sigma\sqrt{2\pi}} e^{-\frac{(ln(cx-a)-\mu)^2}{2\sigma^2}} \tag{68}$$

where $x > 0$. Since we have five unknown parameters, we establish five equations to solve them. Equation one is given by the mean headway. We obtain the mean from the general form of lognormal distribution.

$$\begin{aligned} E(x,\mu,\sigma,a,b,c) &= \int_0^\infty x \frac{b}{(cx-a)\sigma\sqrt{2\pi}} e^{-\frac{(ln(cx-a)-\mu)^2}{2\sigma^2}} dx \\ &= \frac{1}{c}\int_0^\infty \frac{(cx-a)b}{(cx-a)\sigma\sqrt{2\pi}} e^{-\frac{(ln(cx-a)-\mu)^2}{2\sigma^2}} dx \\ &+ \frac{1}{c}\int_0^\infty \frac{ab}{(cx-a)\sigma\sqrt{2\pi}} e^{-\frac{(ln(cx-a)-\mu)^2}{2\sigma^2}} dx \\ &= \frac{b}{c^2} e^{\mu+\sigma^2/2} + \frac{ab}{c^2} \end{aligned} \tag{69}$$

The parameters we applied to simulations are the following:

- The traffic density: 2800 vehicle/hour
- Maximum speed: 60 km/h
- Minimum speed: 20 km/h
- Road: highway
- # of lanes: 2

Total 48257 headway readings are collected. Table 9 shows the simulation results: headway in meters and its frequency. We select four other headways and the probability according to the four points, i.e. (x=8.0, p=0.0100), (x=19.0, p=0.0442),

(x=33.0, p=0.0241), (x=47.0, p=0.0098). Thereafter, five equations are created.

$$\begin{cases} 36.0900 &= \frac{b}{c^2}e^{\mu+\sigma^2/2} + \frac{ab}{c^2} \\ 0.0100 &= \frac{b}{(8.0\cdot c-a)\sigma\sqrt{2\pi}}e^{-\frac{(ln(8.0\cdot c-a)-\mu)^2}{2\sigma^2}} \\ 0.0442 &= \frac{b}{(19.0\cdot c-a)\sigma\sqrt{2\pi}}e^{-\frac{(ln(19.0\cdot c-a)-\mu)^2}{2\sigma^2}} \\ 0.0241 &= \frac{b}{(33.0\cdot c-a)\sigma\sqrt{2\pi}}e^{-\frac{(ln(33.0\cdot c-a)-\mu)^2}{2\sigma^2}} \\ 0.0098 &= \frac{b}{(47.0\cdot c-a)\sigma\sqrt{2\pi}}e^{-\frac{(ln(47.0\cdot c-a)-\mu)^2}{2\sigma^2}} \end{cases} \tag{70}$$

The base 10 logarithm is taken for all the five equations for the sake of programming. With the aid of computer, we can numerically solve the solution of $\mu = 2.95, \sigma = 0.55, a = 2.93, b = 1, c = 0.80$. Different selections of the headway and its frequency might reach to different value of these parameters but they are close to the one we calculated. Therefore, the lognormal form of headway is

$$f(x) = \frac{1}{(0.80x - 2.93)0.55\sqrt{2\pi}}e^{-\frac{(ln(0.80x-2.93)-2.95)^2}{2*0.55^2}} \tag{71}$$

Similarly, a general form of the normal distribution is defined as

$$N(x, \mu, \sigma, a, b, c) = \frac{b}{\sigma\sqrt{2\pi}}e^{-\frac{(cx-a-\mu)^2}{2\sigma^2}} \tag{72}$$

We obtained $\sigma = 9.51, \mu = 0.085, a = 8.55, b = 1.01, c = 0.53$ after selecting five points and solving five equations.

A general form of the exponential distribution is defined as

$$Expo(x, \lambda, a, b, c) = b\lambda e^{-\lambda(cx-a)}. \tag{73}$$

We obtained $\lambda = 2.01, a = 1.05, b = 0.01, c = 0.03$ after selecting four points and solving four equations.

A general form of the gamma distribution is defined as

$$\Gamma(x, \theta, k, a, b, c) = b(cx - a)^{k-1}\frac{e^{-(cx-a)/\theta}}{\theta^k\Gamma(k)}. \tag{74}$$

We obtained $\theta = 4.01, k = 6.085, a = 1.05, b = 1.01, c = 1.03$ after selecting five points and solving five equations.

The comparison of these fitting curves are shown in Figure 13. It is clear that

the log-normal distribution fits the simulation results better than other distributions. Although the gamma distribution has a similar bell-shape curve, the gamma distribution does not match the simulation results except at several points.

TABLE 9: Simulation Headway Values (H) and Frequencies (Pr)

H	Pr	H	Pr	H	Pr
5.0	0.0010	6.0	0.0028	7.0	0.0059
8.0	0.0100	9.0	0.0148	10.0	0.0201
11.0	0.0253	12.0	0.0296	13.0	0.0340
14.0	0.0377	15.0	0.0396	16.0	0.0412
17.0	0.0432	18.0	0.0431	19.0	0.0442
20.0	0.0441	21.0	0.0426	22.0	0.0414
23.0	0.0411	24.0	0.0399	25.0	0.0376
26.0	0.0358	27.0	0.0342	28.0	0.0325
29.0	0.0312	30.0	0.0297	31.0	0.0275
32.0	0.0266	33.0	0.0241	34.0	0.0236
35.0	0.0222	36.0	0.0201	37.0	0.0189
38.0	0.0179	39.0	0.0167	40.0	0.0156
41.0	0.0145	42.0	0.0138	43.0	0.0130
44.0	0.0122	45.0	0.0114	46.0	0.0105
47.0	0.0098	48.0	0.0091	49.0	0.0085
50.0	0.0081	51.0	0.0074	52.0	0.0069
53.0	0.0066	54.0	0.0061	55.0	0.0057
56.0	0.0053	57.0	0.0050	58.0	0.0046
59.0	0.0043	60.0	0.0041	61.0	0.0038
62.0	0.0035	63.0	0.0033	64.0	0.0031

A.2 LINK DURATION CASES

A.2.1 The Same Direction Cases

Case a

In this case, only t_β and t_ε exist, as shown in Figure 20.a. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_i>0>a_j; v_j>v_i>0; 0\le t\le t_\alpha\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i>0>a_j; v_j>v_i>0; t_\alpha<t\le t_\varepsilon\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+X+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & ,\{a_i>0>a_j; v_j>v_i>0; t_\varepsilon<t<t_\beta\} \\ \frac{300+x+\frac{v_m-v_i}{2}t_\varepsilon+\frac{v_j}{2}t_\beta}{v_m} & ,\{a_i>0>a_j; v_j>v_i>0; t_\beta<t\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_j>0>a_i; v_i>v_j>0; 0\le t\le t_\alpha\} \\ \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_j>0>a_i; v_i>v_j>0; t_\alpha<t\le t_\varepsilon\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_j(300-X+\frac{v_m-v_j}{2}t_\varepsilon)}}{a_i} & ,\{a_j>0>a_i; v_i>v_j>0; t_\varepsilon<t\le t_\beta\} \\ \frac{300-X+\frac{v_m-v_j}{2}t_\varepsilon+\frac{v_i}{2}t_\beta}{v_m} & ,\{a_j>0>a_i; v_i>v_j>0; t_\beta<t\} \end{cases}$$

Case b

In this case, the difference from case a in Figure 20.b is the order of t_β and t_ε. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_i>0>a_j; v_j>v_i>0; 0\le t\le t_\alpha\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i>0>a_j; v_j>v_i>0; t_\alpha<t\le t_\beta\} \\ \frac{-v_i+\sqrt{v_i^2+2a_i(300+X+S_j(t_\beta))}}{a_i} & ,\{a_i>0>a_j; v_j>v_i>0; t_\beta<t<t_\varepsilon\} \\ \frac{300+x+\frac{v_m-v_i}{2}t_\varepsilon+\frac{v_j}{2}t_\beta}{v_m} & ,\{a_i>0>a_j; v_j>v_i>0; t_\varepsilon<t\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_j>0>a_i; v_i>v_j>0; 0\le t\le t_\alpha\} \\ \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_j>0>a_i; v_i>v_j>0; t_\alpha<t\le t_\beta\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X-S_i(t_\beta))}}{a_j} & ,\{a_j>0>a_i; v_i>v_j>0; t_\beta<t\le t_\varepsilon\} \\ \frac{300-X+\frac{v_m-v_j}{2}t_\varepsilon+\frac{v_i}{2}t_\beta}{v_m} & ,\{a_j>0>a_i; v_i>v_j>0; t_\varepsilon<t\} \end{cases}$$

Case c

In this case, the difference from case b in Figure 20.c is the intersection of two velocity curves. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_j > 0 > a_i; v_j > v_i > 0; 0 \le t \le t_\beta\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X+S_i(t_\beta))}}{a_j} & ,\{a_j > 0 > a_i; v_j > v_i > 0; t_\beta < t < t_\varepsilon\} \\ \frac{300-X+\frac{v_i}{2}t_\beta+\frac{v_m-v_j}{2}t_\varepsilon}{v_m} & ,\{a_j > 0 > a_i; v_j > v_i > 0; t_\varepsilon < t\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i > 0 > a_j; v_i > v_j > 0; 0 \le t \le t_\beta\} \\ \frac{-v_i+\sqrt{v_i^2+2a_i(300+X+S_j(t_\beta))}}{a_i} & ,\{a_j > 0 > a_i; v_i > v_j > 0; t_\beta < t < t_\varepsilon\} \\ \frac{300+X+\frac{v_j}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{a_j > 0 > a_i; v_i > v_j > 0; t_\varepsilon < t\} \end{cases}$$

Case d

In this case, the difference from case c in Figure 20.d is the order of t_β and t_ε. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_j > 0 > a_i; v_j > v_i > 0; 0 \le t \le t_\varepsilon\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_i(300-X+\frac{v_m-v_j}{2}t_\varepsilon)}}{a_i} & ,\{a_j > 0 > a_i; v_j > v_i > 0; t_\varepsilon < t \le t_\beta\} \\ \frac{300-X+\frac{v_i}{2}t_\beta+\frac{v_m-v_j}{2}t_\varepsilon}{v_m} & ,\{a_j > 0 > a_i; v_j > v_i > 0; t_\beta < t\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i > 0 > a_j; v_i > v_j > 0; 0 \le t \le t_\varepsilon\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+X+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & ,\{a_j > 0 > a_i; v_i > v_j > 0; t_\varepsilon < t \le t_\beta\} \\ \frac{300+X+\frac{v_j}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{a_j > 0 > a_i; v_i > v_j > 0; t_\beta < t\} \end{cases}$$

Case e

In this case, the difference from case b in Figure 20.e is that both acceleration are positive. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_i > a_j > 0; v_j > v_i > 0; 0 \le t \le t_\alpha\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i > a_j > 0; v_j > v_i > 0; t_\alpha < t \le t_\varepsilon\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+x+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & ,\{a_i > a_j > 0; v_j > v_i > 0; t_\varepsilon < t \le t_\zeta\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_j > a_i > 0; v_i > v_j > 0; 0 \le t \le t_\alpha\} \\ \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_j > a_i > 0; v_i > v_j > 0; t_\alpha \le t\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_i(300-x+\frac{v_m-v_j}{2}t_\varepsilon)}}{a_i} & ,\{a_j > a_i > 0; v_i > v_j > 0; t_\varepsilon < t \le t_\zeta\} \\ +\infty & ,\{otherwise\} \end{cases}$$

Case f

In this case, the difference from case e in Figure 20.f is the intersection of two velocity curves. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{a_i, a_j > 0; v_j > v_i > 0; 0 < t \le t_\varepsilon\} \\ \frac{300-X}{v_r} & ,\{a_i = a_j > 0; v_j > v_i > 0\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_i(300-X+\frac{v_m-v_j}{2}t_\varepsilon)}}{a_i} & \{a_i, a_j > 0; v_j > v_i > 0; t_\varepsilon < t \le t_\zeta\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{a_i, a_j > 0; v_i > v_j > 0\} \\ \frac{300+X}{v_r} & ,\{a_j = a_i > 0; v_i > v_j > 0\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+X+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & \{a_i, a_j > 0; v_i > v_j > 0; t_\varepsilon < t \le t_\zeta\} \\ \infty & ,\{otherwise\} \end{cases}$$

A.2.2 The Oncoming Direction Cases

case a

As shown in Figure 21.a, we can obtain the link duration in this case.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_i(300-X+\frac{v_m-v_j}{2}t_\varepsilon)}}{a_i} & ,\{t_\varepsilon \le t \le t_\beta; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{300-X+\frac{v_i}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{t_\beta < t; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_i > 0 > v_j; a_i, a_j > 0\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+X+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & ,\{t_\varepsilon \le t \le t_\beta; v_i > 0 > v_j; a_i, a_j > 0\} \\ \frac{300+X+\frac{v_j}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{t_\beta < t; v_i > 0 > v_j; a_i, a_j > 0\} \end{cases}$$

case b

In this case, the difference from case a in Figure 21.b is the order of t_β and t_ε. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\beta; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X+S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\varepsilon; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{300-X+\frac{v_i}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{t_\varepsilon < t; v_j > 0 > v_i; a_i, a_j > 0\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\beta; v_i > 0 > v_j; a_i, a_j > 0\} \\ \frac{-v_i+\sqrt{v_i^2+2a_i(300+X+S_j(t_\beta))}}{a_i} & ,\{t_\beta < t \le t_\varepsilon; v_i > 0 > v_j; a_i, a_j > 0\} \\ \frac{300+X+\frac{v_j}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{t_\varepsilon < t; v_i > 0 > v_j; a_i, a_j > 0\} \end{cases}$$

case c

In this case, the difference from case a in Figure 21.c is the directions of two velocity. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{-(v_j+v_m)+\sqrt{(v_j+v_m)^2+2a_j(300-X-\frac{v_m+v_i}{2}t_\varepsilon)}}{a_j} & ,\{t_\varepsilon < t \le t_\beta; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{300-X-\frac{v_j}{2}t_\beta+\frac{v_m+v_i}{2}t_\varepsilon}{v_m} & ,\{t_\beta < t; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_i > 0 > v_j; a_i, a_j < 0\} \\ \frac{-(v_i+v_m)+\sqrt{(v_i+v_m)^2+2a_i(300+X+\frac{v_m+v_i}{2}t_\varepsilon)}}{a_j} & ,\{t_\varepsilon < t \le t_\beta; v_i > 0 > v_j; a_i, a_j < 0\} \\ \frac{300+X+\frac{v_j}{2}t_\beta+\frac{v_m-v_i}{2}t_\varepsilon}{v_m} & ,\{t_\varepsilon < t; v_i > 0 > v_j; a_i, a_j < 0\} \end{cases}$$

case d

In this case, the difference from case c in Figure 21.d is the order of t_β and t_ε. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\beta; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{-v_i-\sqrt{v_i^2-2a_i(300-X-S_j(t_\beta))}}{a_i} & ,\{t_\beta < t \le t_\varepsilon; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{300-X-\frac{v_j}{2}t_\beta+\frac{v_m+v_i}{2}t_\varepsilon}{v_m} & ,\{t_\varepsilon < t; v_j > 0 > v_i; a_i, a_j < 0\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\beta; v_i > 0 > v_j; a_i, a_j < 0\} \\ \frac{-v_j-\sqrt{v_j^2-2a_j(300+X-S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\varepsilon; v_i > 0 > v_j; a_i, a_j < 0\} \\ \frac{300+X-\frac{v_i}{2}t_\beta+\frac{v_m+v_j}{2}t_\varepsilon}{v_m} & ,\{t_\varepsilon < t; v_i > 0 > v_j; a_i, a_j < 0\} \end{cases}$$

case e

In this case, the difference from case d in Figure 21.e is the accelerations of two vehicles. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_i > 0 > v_j; a_i > 0 > a_j\} \\ \frac{-(v_i-v_m)-\sqrt{(v_i-v_m)^2-2a_i(300-X+\frac{v_m+v_j}{2}t_\varepsilon)}}{a_i} & ,\{t_\varepsilon \le t \le t_\zeta; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{300-X+t_\zeta(v_m-v_j)/2+t_\varepsilon(v_m+v_i)}{2v_m} & ,\{t_\zeta < t; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{-(v_j-v_m)-\sqrt{(v_j-v_m)^2-2a_j(300+x+\frac{v_m-v_i}{2}t_\varepsilon)}}{a_j} & ,\{t_\varepsilon \le t \le t_\zeta; v_i > 0 > v_j; a_i > 0 > a_j\} \\ \frac{300+X-t_\varepsilon(v_m+v_j)/2-t_\zeta(v_m-v_i)}{2v_m} & ,\{t_\zeta < t; v_i > 0 > v_j; a_i > 0 > a_j\} \end{cases}$$

case f

In this case, the difference from case e in Figure 21.f is on which direction vehicles reach speed limit first. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{-(v_j+v_m)+\sqrt{(v_j+v_m)^2+2a_j(300-X+\frac{v_m+v_i}{2}t_\varepsilon)}}{a_j} & ,\{t_\varepsilon \le t \le t_\zeta; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{300-X+t_\varepsilon(v_m+v_i)/2-t_\zeta(v_m-v_j)}{2v_m} & ,\{t_\zeta < t; v_j > 0 > v_i; a_j > 0 > a_i\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\varepsilon; v_i > 0 > v_j; a_i > 0 > a_j\} \\ \frac{-(v_i+v_m)+\sqrt{(v_i+v_m)^2+2a_i(300+X+\frac{v_m+v_j}{2}t_\varepsilon)}}{a_i} & ,\{t_\varepsilon \le t \le t_\zeta; v_i > 0 > v_j; a_i > 0 > a_j\} \\ \frac{300+X-t_\zeta(v_m-v_i)/2-t_\varepsilon(v_m+v_j)}{2v_m} & ,\{t_\zeta < t; v_i > 0 > v_j; a_i > 0 > a_j\} \end{cases}$$

A.2.3 The Decelerating Cases

Case a

As shown in Figure 22.a, both vehicles are decelerating and we can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\alpha; v_j > v_i > 0; 0 > a_i > a_j\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{t_\alpha < t \le t_\beta; v_j > v_i > 0; 0 > a_i > a_j\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300+X+S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\gamma; v_j > v_i > 0; 0 > a_i > a_j\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\alpha; v_i > v_j > 0; 0 > a_j > a_i\} \\ \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{t_\alpha < t \le t_\beta; v_i > v_j > 0; 0 > a_j > a_i\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X-S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\gamma; v_i > v_j > 0; 0 > a_j > a_i\} \\ +\infty & ,\{t_\gamma < t; \} \end{cases}$$

Case b

In this case, the difference from case a in Figure 22.b is the intersection of the two velocity curves. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\beta; v_j > v_i > 0; a_j \neq a_i < 0\} \\ \frac{300-X}{v_r} & ,\{0 \le t \le t_\beta; v_j > v_i > 0; a_j = a_i <= 0\} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X-S_i(t_\beta))}}{a_j} & ,\{t_\beta < t; v_j > v_i > 0; a_j, a_i < 0\} \\ \frac{300-X-S_i(t_\beta)}{v_j+a_j t\beta} & ,\{t_\beta < t; v_j > v_i > 0; a_j = 0; a_i <= 0\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\beta; v_i > v_j > 0; a_j \neq a_i < 0\} \\ \frac{300+X}{-v_r} & ,\{0 \le t \le t_\beta; v_i > v_j > 0; a_j = a_i <= 0\} \\ \frac{-v_i+\sqrt{v_i^2+2a_i(300+X+S_j(t_\beta))}}{a_i} & ,\{t_\beta < t; v_i > v_j > 0; a_j, a_i < 0\} \\ \frac{300+X+S_j(t_\beta)}{-(v_i+a_i t\beta)} & ,\{t_\beta < t; v_i > v_j > 0; a_j <= 0; a_i = 0\} \end{cases}$$

Case c

In this case, the difference from case a in Figure 22.c is the direction of the two velocity of the two vehicles. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\beta; v_j > 0 > v_i; a_i > 0 > a_j\} \\ +\infty & ,\{t_\gamma < t; \} \\ \frac{-v_j+\sqrt{v_j^2+2a_j(300-X+S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\gamma; v_j > 0 > v_i; a_i > 0 > a_j\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\beta; v_i > 0 > v_j; a_j > 0 > a_i\} \\ \frac{-v_i+\sqrt{v_i^2+2a_i(300+X+S_j(t_\beta))}}{a_i} & ,\{t_\beta < t \le t_\gamma; v_i > 0 > v_j; a_j > 0 > a_i\} \\ +\infty & ,\{t_\gamma < t; \} \end{cases}$$

Case d

In this case, the difference from case a in Figure 22.d is the direction of the two velocity of the two vehicles. We can obtain the link duration time t.

$$t = \begin{cases} \frac{-v_r+\sqrt{v_r^2+2a_r(300-X)}}{a_r} & ,\{0 \le t \le t_\beta; v_j > 0 > v_i; a_i > 0 > a_j\} \\ +\infty & ,\{t_\gamma < t;\} \\ \frac{-v_i-\sqrt{v_i^2-2a_i(300-X-S_j(t_\beta))}}{a_i} & ,\{t_\beta < t \le t_\gamma; v_j > 0 > v_i; a_i > 0 > a_j\} \\ \frac{-v_r-\sqrt{v_r^2-2a_r(300+X)}}{a_r} & ,\{0 \le t \le t_\beta; v_i > 0 > v_j; a_j > 0 > a_i\} \\ \frac{-v_j-\sqrt{v_j^2-2a_j(300+X-S_i(t_\beta))}}{a_j} & ,\{t_\beta < t \le t_\gamma; v_i > 0 > v_j; a_j > 0 > a_i\} \\ +\infty & ,\{t_\gamma < t;\} \end{cases}$$

VITA

Gongjun Yan
Department of Computer Science
Old Dominion University
Norfolk, VA 23529

Gongjun Yan received his MS in Computer Science from University of Electronics Science and Technology of China in 2004 and BS in Mechanical Engineering in Sichuan Institute of Technology in 1999. Currently, he is a Ph.D candidate in Computer Science at Old Dominion University. His main research areas include wireless security and Intelligent Transportation Systems. In the area of security Gongjun is mostly interested in location security, confidentiality, and availability (the so-called CIA model).

Typeset using LaTeX.

BIBLIOGRAPHY

[1] E. Schoch, F. Kargl, T. Leinmüller, and M. Weber, "Communication patterns in VANETs," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 119–125, November 2008.

[2] L. Scott and D. E. Denning, "Location based encryption technique and some of its applications," in *Proceedings of Institute of Navigation National Technical Meeting 2003*, Anaheim, CA, January 22-24 2003, pp. 734–740.

[3] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Securing sensor networks with location-based keys," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2005)*, New Orleans, LA, 02 May 2005, pp. 1909 – 1914.

[4] D.P. Agrawal and Q.-A. Zeng, *Introduction to Wireless and Mobile Systems*. Brooks/Cole Publishing, 2002.

[5] Car 2 Car Communication Consortium, http://www.car-to-car.org/.

[6] US Department of Transportation, National Highway Traffic Safety Administration, "Vehicle safety communications consortium," http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/CAMP3/pages/VSCC.htm.

[7] A. Takahashi and N. Asanuma, "Introduction of Honda ASV-2 (Advanced Safety Vehicle-Phase 2)," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, Detroit, USA, Oct. 2000, pp. 694–701.

[8] R. Möbus, M. Baotic, and M. Morari, "Multi-Object Adaptive Cruise Control," *Hybrid Systems: Computation and Control*, vol. 2623, pp. 359–374, Apr. 2003.

[9] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security and Privacy Magazine*, vol. 2, no. 3, pp. 49–55, 2004.

[10] US Department of Transportation, "Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems - 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications," ASTM E2213-03, Aug. 2003.

[11] IEEE, "Status of Project IEEE 802.11p," http://grouper.ieee.org/groups/802/11/Reports/tgp_update.htm, Nov. 2006.

[12] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Proceedings of the IEEE Vehicular Technology Conference - Spring*, May 2008, pp. 2036–2040.

[13] Sensor Technologies and Systems, "Forward looking vehicle radar system (FLVRS)," http://www.sensor-tech.com/sub\%20pages/products/AUTOMOTIVE/flvrs.html, 2006.

[14] Toyota, "Pre-crash safety," http://www.toyota.co.jp/en/about_toyota/in_the_world/pdf2007/safety.pdf, 2007.

[15] Electronic License Plate, "Electronic license plate," http://www.identecsolutions.com/electroniclicenseplate.html.

[16] P. Bahl and V. N. Padmanabhan, "Radar: A in-building RF-based user location and tracking system," in *Proceedings of the IEEE Infocom*, Tel Aviv, Israel, MAR 2000, pp. 775–784.

[17] S. Spors, R. Rabenstein, and N. Strobel, "A multi-sensor object localization system," in *VMV '01: Proceedings of the Vision Modeling and Visualization Conference 2001.* Aka GmbH, 2001, pp. 19–26.

[18] K. Chadha, "The global positioning system: challenges in bringing GPS to mainstream consumers," in *Solid-State Circuits Conference, 1998. Digest of Technical Papers. 1998 IEEE International*, San Francisco, CA, pp. 26–28.

[19] C. Harsch, A. Festag, and P. Papadimitratos, "Secure position-based routing for VANETs," *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pp. 26–30, 30 2007-Oct. 3 2007.

[20] J. Douceur, "The Sybil attack," *Lecture Notes in Computer Science: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, vol. 2429, pp. 251–260, 2002.

[21] P. Papadimitratos, V. Gligor, and J. Hubaux, "Securing vehicular communications - assumptions, requirements, and principles," in *Proceedings of Fourth*

Workshop on Embedded Security in Cars (ESCAR), Berlin, Germany, November 2006.

[22] Federal Information Processing Standards (FIPS) in National Institute of Standards and Technology, "Minimum security requirements for federal information and information systems," http://csrc.nist.gov/publications/fips/fips200/FIPS-200-final-march.pdf.

[23] L. Volonino and S. R. Robinson, *Principles and Practice of Information Security*. Pearson Education, 2003.

[24] M. Horton and C. Mugge, *HackNotes Network Security Portable Reference*. New York, NY, USA: McGraw-Hill, Inc., 2003.

[25] C. Adams and S. Lloyd, *Understanding PKI: Concepts, Standards, and Deployment Considerations*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[26] M. Raya, A. Aziz, and J.-P. Hubaux, "Efficient Secure Aggregation in VANETs," in *Proceedings of the ACM Workshop on Vehicular Ad Hoc Networks (VANET)*, Los Angeles, CA, Sep. 2006, pp. 67–75.

[27] S. Capkun and J.-P. Hubaux, "Secure positioning of wireless devices with application to sensor networks," in *Proceedings of IEEE INFOCOM*, vol. 3, Mar. 2005, pp. 1917–1928.

[28] F. Armknecht, A. Festag, D. Westhoff, and K. Zeng, "Cross-layer privacy enhancement and non-repudiation in vehicular communication," in *Proceedings of the Workshop on Mobile Ad-Hoc Networks (WMAN)*, Bern, Switzerland, Mar. 2007.

[29] J. Y. Choi, P. Golle, and M. Jakobsson, "Tamper-evident digital signatures: Protecting certification authorities against malware," in *Proceedings of the IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC)*, 2006, pp. 37–44.

[30] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Proceedings of ACM HotNets*, Nov. 2005.

[31] K. Plößl, T. Nowey, and C. Mletzko, "Towards a security architecture for vehicular ad hoc networks," in *Proceedings of the International Conference on Availability, Reliability and Security (ARES)*, Washington, DC, USA, 2006, pp. 374–381.

[32] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC Experimental 3561, July 2003. [Online]. Available: http://rfc.net/rfc3561.txt

[33] D. B. Johnson, D. A. Maltz, and Y. C. Hu, "The dynamic source routing protocol for mobile ad hoc networks (dsr)," Published Online, IETF MANET Working Group, Tech. Rep., February 2007.

[34] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," *SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, pp. 234–244, 1994.

[35] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *Communications Magazine, IEEE*, vol. 44, no. 1, pp. 74–82, Jan. 2006.

[36] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *Mob. Netw. Appl.*, vol. 1, no. 2, pp. 183–197, 1996.

[37] O. Abedi, M. Fathy, and J. Taghiloo, "Enhancing AODV routing protocol using mobility parameters in VANET," in *AICCSA '08: Proceedings of the 2008 IEEE/ACS International Conference on Computer Systems and Applications*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 229–235.

[38] X. Li and L. Cuthbert, "On-demand node-disjoint multipath routing in wireless ad hoc network," in *LCN '04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 419–420.

[39] V. Namboodiri and L. Gao, "Prediction-based routing for vehicular ad hoc networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 4, pp. 2332–2345, July 2007.

[40] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support its services in VANET networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3337–3347, Nov. 2007.

[41] H. F. Wedde, S. Lehnhoff, and B. van Bonn, "Highly dynamic and scalable VANET routing for avoiding traffic congestions," in *VANET '07: Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*. New York, NY, USA: ACM, 2007, pp. 81–82.

[42] Z. Niu, W. Yao, Q. Ni, and Y. Song, "DeReQ: a QoS routing algorithm for multimedia communications in vehicular ad hoc networks," in *IWCMC '07: Proceedings of the 2007 international conference on Wireless communications and mobile computing*. New York, NY, USA: ACM, 2007, pp. 393–398.

[43] R. He, H. Rutagemwa, and X. Shen, "Differentiated reliable routing in hybrid vehicular ad-hoc networks," May 2008, pp. 2353–2358.

[44] H. Kim, J. Paik, B. Lee, and D. Lee, "SARC: A street-based anonymous vehicular ad hoc routing protocol for city environment," in *EUC '08: Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 324–329.

[45] T. Kitani, T. Shinkawa, N. Shibata, K. Yasumoto, M. Ito, and T. Higashino, "Efficient VANET-based traffic information sharing using buses on regular routes," May 2008, pp. 3031–3036.

[46] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. Decouto, "CarNet: A scalable ad hoc wireless network system," in *In Proceedings of the 9th ACM SIGOPS European workshop: Beyond the PC: New Challenges for the Operating System*. ACM Press, 2000, pp. 61–65.

[47] T. Kato, K. Kadowaki, T. Koita, and K. Sato, "Routing and address assignment using lane/position information in a vehicular ad hoc network," in *APSCC '08: Proceedings of the 2008 IEEE Asia-Pacific Services Computing Conference*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 1600–1605.

[48] J. Bronsted and L. M. Kristensen, "Specification and performance evaluation of two zone dissemination protocols for vehicular ad-hoc networks," in *ANSS '06: Proceedings of the 39th annual Symposium on Simulation.* Washington, DC, USA: IEEE Computer Society, 2006, pp. 68–79.

[49] J. Gong, C.-Z. Xu, and J. Holle, "Predictive directional greedy routing in vehicular ad hoc networks," in *ICDCSW '07: Proceedings of the 27th International Conference on Distributed Computing Systems Workshops.* Washington, DC, USA: IEEE Computer Society, 2007, p. 2.

[50] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve, "A routing strategy for vehicular ad hoc networks in city environments," June 2003, pp. 156–161.

[51] M. Kihl, M. Sichitiu, T. Ekeroth, and M. Rozenberg, *Reliable Geographical Multicast Routing in Vehicular Ad-Hoc Networks.* Springer Berlin / Heidelberg, 2007.

[52] S. Momeni and M. Fathy, "VANET's communication," in *Spread Spectrum Techniques and Applications, 2008. ISSSTA '08. IEEE 10th International Symposium on*, Aug. 2008, pp. 608–612.

[53] G. Yan and S. Olariu, "A reliable routing in vehicular ad hoc networks using probabilistic analysis of path stability," *Technique Report in Computer Science of Old Dominion University*, 2009.

[54] W. Sun, H. Yamaguchi, and K. Yukimasa, "GVGrid: A QoS routing protocol for vehicular ad hoc networks," in *Proceedings of the Fourteenth IEEE International Workshop on Quality of Service (IWQoS 2006)*, Yale University, New Haven, CT, USA, June 19-21 2006, pp. 130–139.

[55] Q. Yang, A. Lim, and P. Agrawal, "Connectivity aware routing in vehicular networks," 31 2008-April 3 2008, pp. 2218–2223.

[56] H. Jiang, H. Guo, and L. Chen, "Reliable and efficient alarm message routing in VANET," in *ICDCSW '08: Proceedings of the 2008 The 28th International Conference on Distributed Computing Systems Workshops.* Washington, DC, USA: IEEE Computer Society, 2008, pp. 186–191.

[57] A. Bachir and A. Benslimane, "A multicast protocol in ad hoc networks inter-vehicle geocast," in *Proc. 57th IEEE Semiannual Vehicular Technology Conference*, vol. 4, 2003, pp. 2456–2460.

[58] K. Ibrahim and M. C. Weigle, "Optimizing CASCADE data aggregation for VANETs," in *Proceedings of the International Workshop on Mobile Vehicular Networks (MoVeNet)*, Atlanta, GA, sep 2008, pp. 724–729.

[59] K. Ibrahim, M. C. Weigle, and M. Abuelela, "p-IVG: Probabilistic inter-vehicle geocast for dense vehicular networks," in *Proceedings of the IEEE Vehicular Technology Conference - Spring*, Barcelona, Spain, April 2009.

[60] I. Rubin and Y.-C. Liu, "Link stability models for QoS ad hoc routing algorithms," vol. 5, Oct. 2003, pp. 3084–3088 Vol.5.

[61] S. Jiang, D. He, and J. Rao, "A prediction-based link availability estimation for mobile ad hoc networks," vol. 3, 2001, pp. 1745–1752 vol.3.

[62] Y.-C. Cheng and T. Robertazzi, "Critical connectivity phenomena in multihop radio models," *Communications, IEEE Transactions on*, vol. 37, no. 7, pp. 770–777, Jul 1989.

[63] P. Piret, "On the connectivity of radio networks," *Information Theory, IEEE Transactions on*, vol. 37, no. 5, pp. 1490–1492, Sep 1991.

[64] P. Gupta and P. Kumar, "Critical power for asymptotic connectivity," *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, vol. 1, pp. 1106–1110 vol.1, 1998.

[65] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," in *MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*. New York, NY, USA: ACM, 2002, pp. 80–91.

[66] P. Santi and D. M. Blough, "The critical transmitting range for connectivity in sparse wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 25–39, 2003.

[67] C. Bettstetter and C. Hartmann, "Connectivity of wireless multihop networks in a shadow fading environment," *Wirel. Netw.*, vol. 11, no. 5, pp. 571–579, 2005.

[68] M. Nekovee and M. Ko, "Throughput analysis of Wi-Fi based broadband access for mobile users on the highway," in *Proceedings of 13-th IEEE International Conference on Networks*, Boston, Massachusetts, USA, 16-18 Nov 2005, p. 6.

[69] M. Nekovee, "Modeling the spread of worm epidemics in vehicular ad hoc networks," in *Proceedings of the 63-th IEEE Vehicular Technology Conference VTC Spring 2006*, Melbourne, Australia, 7-10 May 2006, pp. 841–845.

[70] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Improved security in geographic ad hoc routing through autonomous position verification," in *Proceedings of the ACM Workshop on Vehicular Ad Hoc Networks (VANET)*, Los Angeles, CA, 2006, pp. 57–66.

[71] T. Suen and A. Yasinsac, "Ad hoc network security: Peer identification and authentication using signal properties," in *Systems, Man and Cybernetics (SMC) Information Assurance Workshop*, June 2005, pp. 432–433.

[72] B. Xiao, B. Yu, and C. Gao, "Detection and localization of Sybil nodes in VANETs," in *Proceedings of the Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks*, Los Angeles, CA, USA, 2006, pp. 1–8.

[73] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil attack in sensor networks: analysis & defenses," in *Proceedings of International Symposium on Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, 2004, pp. 259–268.

[74] C. Piro, C. Shields, and B. N. Levine, "Detecting the Sybil attack in mobile ad hoc network," in *Proceedings of the International Conference on Security and Privacy in Communication Networks*, Aug. 2006, pp. 1–11.

[75] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications," *IEEE Wireless Communications Magazine*, pp. 8–15, Oct. 2006.

[76] K. P. Laberteaux, J. J. Haas, and Y.-C. Hu, "Security certificate revocation list distribution for VANET," in *VANET '08: Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, 2008, pp. 88–89.

[77] G. Yan, S. Olariu, and M. C. Weigle, "Providing VANET security through active position detection," *Computer Communications: Special Issue on Mobility Protocols for ITS VANET*, vol. 31, no. 12, pp. 2883–2897, Jul 2008.

[78] R. Ramesh and S. Kumar, "Secure position routing using ad hoc network," Dec. 2006, pp. 200–201.

[79] D. Denning and P. MacDoran, "Location-based authentication: Grounding cyberspace for better security," *Computer Fraud and Security*, vol. 1996, no. 2, pp. 12–16, 1996.

[80] A. Al-Fuqaha and O. Al-Ibrahim, "Geo-encryption protocol for mobile networks," *Comput. Commun.*, vol. 30, no. 11-12, pp. 2510–2517, 2007.

[81] P. Reddy, K.R.Sudha, and P. S. Naidu, "A modified location-dependent image encryption for mobile information system," *International Journal of Engineering Science and Technology*, vol. 2, no. 5, pp. 1060–1065, 2010.

[82] P. Reddy, K.R.Sudha, and S. Rao, "Data encryption technique using location based key dependent permutation and circular rotation," *International Journal of Computer and Network Security (IJCNS)*, vol. 2, no. 3, pp. 46–50, 2010.

[83] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security.* New York, NY, USA: ACM, 2002, pp. 41–47.

[84] Y. Cho and L. Bao, "Secure access control for location-based applications in WLAN systems," in *Proceedings of 2006 IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2006, pp. 852–857.

[85] I. Greenberg, "The log-normal distribution of headways," *Australian Road Research*, vol. 2, no. 7, pp. 14–18, 1966.

[86] T. S. Rappaport, *Wireless communications principles and practices.* Prentice-Hall, 2002.

[87] R. B. (Editor), *Handbook of Engineering Electromagnetics (Hardcover)*. CRC, September 1, 2004.

[88] K. Matsunaga, "Insufficient headway and unforseen greater stopping distance as combined factors in traffic accidents," *Proceedings of First Japan-Finland Joint Meeting on Traffic Safety*, August 1996.

[89] S. F. Sheet, "Headway times and road safety," September 2007.

[90] A. Y. Abul-Magd, "Modeling highway-traffic headway distributions using superstatistics," *Physical Review Series E*, vol. 76, no. 5, pp. 057–101, Dec 2007.

[91] R. J. Cowan, "Useful headway models," *Transportation Research*, vol. 9, no. 6, pp. 371–375, 1975.

[92] O. C. Puan, "Driverś car following headway on single carriageway roads," *Malaysian Journal of Civil Engineering (MJCE)*, vol. 16, no. 2, pp. 15–27, Jan 2004.

[93] A. Tomoeda, K. Nishinari, D. Chowdhury, and A. Schadschneider, "An information-based traffic control in a public conveyance system: reduced clustering and enhanced efficiency," *Physica A*, vol. 384, pp. 600–612, 2007.

[94] D. Chowdhury, K. Ghosh, A. Majumdar, S. Sinha, and R. B. Stinchcombe, "Particle-hopping models of vehicular traffic: Distributions of distance headways and distance between jams," *Physica A*, vol. 246, pp. 471–486(16), 1 December 1997.

[95] J. D. Griffiths and J. G. Hunt, "Vehicle headways in urban areas," *Traffic Engineering Control*, vol. 32, no. 10, pp. 458–462, 1991.

[96] M. Krbálek and P. Šeba, "Headway statistics of public transport in mexican cities," *Journal of Physics A: Mathematical and General*, vol. 36, no. 2, pp. L7–L11, May 2003.

[97] C. Beck and E. G. Cohen, "Superstatistics," *Physica A*, vol. 322, no. 1, pp. 267–275, May 2003.

[98] M. Treiber, "Microsimulation of road traffic," http://www.traffic-simulation.de, 2005.

[99] J. S. Seybold, *Introduction to RF propagation*. Wiley, 2005.

[100] M. Feuerstein, K. L. Blackard, T. S. Rappaport, S. Y. Seidel, and H. H. Xia, "Path loss, delay spread, and outage models as functions of antenna height for microcellular system design," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 3, pp. 487–498, August 1994.

[101] L. Fenton, "The sum of log-normal probability distributions in scatter transmission systems," *Communications, IEEE Transactions on [legacy, pre - 1988]*, vol. 8, no. 1, pp. 57–67, 1960.

[102] G. L. Stüber, *Principles of mobile communication (2nd ed.)*. Norwell, MA, USA: Kluwer Academic Publishers, 2001.

[103] N. Beaulieu and Q. Xie, "An optimal log-normal approximation to log-normal sum distributions," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 2, pp. 479–489, March 2004.

[104] Z. Liu, J. Almhana, and R. McGorman, "Approximating log-normal sum distributions with power log-normal distribution," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 4, pp. 2611–2617, July 2008.

[105] G. Yan and S. Olariu, "A probabilistic analysis of link stability in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, 2010 (to appear).

[106] T. Leinmüller and E. Schoch, "Greedy routing in highway scenarios: The impact of position faking nodes," in *Proceedings of the Workshop on Intelligent Transportation (WIT)*, Mar. 2006.

[107] T. Leinmüller and E. Schoch and F. Kargl and C. Maihöfer, "Influence of falsified position data on geographic ad-hoc routing," in *Proceedings of the European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS)*, Jul. 2005.

[108] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wirel. Netw.*, vol. 8, no. 2/3, pp. 153–167, 2002.

[109] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1557–1568, Oct. 2007.

[110] L. Tian, Y. Zhou, and L. Tang, "Improving GPS positioning precision by using optical encoders," in *Proceedings of Intelligent Transportation Systems*, Dearborn, MI, October 2000, pp. 293–298.

[111] P. C. Mahalanobis, "On the generalised distance in statistics," in *Proceedings National Institute of Science, India*, vol. 2, no. 1, April 1936, pp. 49–55. [Online]. Available: http://ir.isical.ac.in/dspace/handle/1/1268

[112] H. Caussinus and A. Ruiz, "Interesting projections of multidimensional data by means of generalized principal component analysis," in *COMPSTAT 90*. Physica-Verlag, 1990, pp. 121–126.

[113] P. Filzmoser, "A multivariate outlier detection method," in *Proceedings of the Seventh International Conference on Computer Data Analysis and Modeling*, Belarusian State University, Minsk, 2004, pp. 18–22.

[114] J. N.G. Terry, "How to read the universal transverse mercator (*utm*) grid," *GPS World*, April 1996, pp. 32.

[115] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, p. 1805, 2000.

[116] D. Molkdar, "Review on radio propagation into and within buildings," *IEE Proceedings on Microwaves, Antennas and Propagation*, vol. 138, no. 1, pp. 61–73, Feb 1991.

[117] G. Delisle, J.-P. Lefevre, M. Lecours, and J.-Y. Chouinard, "Propagation loss prediction: A comparative study with application to the mobile radio channel," *IEEE Transactions on Vehicular Technology*, vol. 34, no. 2, pp. 86–96, May 1985.

[118] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support its services in VANET networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3337–3347, Nov. 2007.

[119] Widyawan, M. Klepal, and D. Pesch, "Influence of predicted and measured fingerprint on the accuracy of RSSI-based indoor location systems," in *Positioning, Navigation and Communication, 2007. WPNC '07. 4th Workshop on*, March 2007, pp. 145–151.

[120] M. Porretta, P. Nepa, G. Manara, and F. Giannetti, "Location, location, location," *Vehicular Technology Magazine, IEEE*, vol. 3, no. 2, pp. 20–29, June 2008.

[121] H. Thode, *Testing for Normality; electronic version.* Abingdon: Dekker, 2002.

[122] Open source, "Simulation of urban mobility," http://sumo.sourceforge.net.

[123] "The Network Simulator NS-2," http://www.isi.edu/nsnam/ns/.