JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

# Enhancing Intra Safety Area Queuing (ISAQ) for Channel-Aware based BSM Verification Scheme in VANETS

BY

Naol Getachew

Advisor: Mr. Kebebew Ababu(Ass.Prof.)

Co-Advisor: Mr. Samuel Sisay(M.Sc.)

THESIS SUBMITTED TO SCHOOL OF COMPUTING OF JIMMA UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER NETWORKING

January 2021

Jimma, Ethiopia

**Approval sheet**

This Independent Research entitled "Enhancing Intra Safety Area Queuing (ISAQ) for Channel-Aware based BSM Verification Scheme in VANETS" has been read and approved as meeting the preliminary research requirements of the School of Computing in partial fulfillment for the award of the degree of Master in Computer Networking,

Jimma University, Jimma, Ethiopia.

Mr. Kebebew Ababu (Assist. Professor)

Principal Advisor

Mr. Samuel Sisay (MSc.)

Co-Advisor

Principal Investigator

Institute coordinator for Research, P, GS

## DEDICATION

I dedicate this thesis to my family!

**Acknowledgment**

The one above all, I would like to thank God Almighty for giving me the strength, knowledge, ability and opportunity to undertake this thesis study and to preserve and complete it.

I would also like to thank my study place, Jimma Institute of Technology, Jimma University for the opportunity they gave me to study here. I would like to express my deep gratitude to my research principal advisor, Mr. Kebebew Ababu (Asst. Prof.)**.** Without his assistance and dedicated involvement in every step throughout the process, this thesis would have never been accomplished. No word! Just thank you very much. I would also like to thank very much my co-advisor Mr. Samuel Sisay (MSc.). My deep gratitude also to Mr. Getamesay Haile (MSc.) and Mr. Gemechu Birhanu (MSc.) for their patient guidance, enthusiastic encouragement, and useful critiques of this research work.

Most importantly, none of this could have happened without my family. I wish to thank my parents for their support and encouragement throughout my study.

Mr. Naol Getachew

# List of Figures

## List of Tables

# Acronyms, abbreviation

## Terminologies

Prioritization ---- refers to the activity that arranges items or activities in order of importance relative to each other [13].

Clustering ----refers to the process of organizing objects into groups whose members are similar with respect to a similarity or distance criterion [10].

# Table of Contents

**Contents** **Pages**

## Abstract

One of the main goals of vehicular ad hoc networks (VANETs) is to increase road safety and traffic efficiency, by using information which is shared among vehicles in a wide range of Intelligent Transportation Systems (ITS) applications such as crash warning, sudden-brake warning, and lane-change warning. Safety messages are transmitted from each vehicle at a fixed rate such as ten messages per second. In dense roads with many lanes or with many vehicles close to each other, there could be many safety messages received by a vehicle. Each safety message must be verified by a time consuming cryptographic operation before its information can be reliably utilized. This leads to a problem since the rate that messages are received can be much higher than the verification rate. This problem could be serious with densely occupied roads and when ITS applications require a high transmission rate of safety messages.

To solve this problem, some solutions have been proposed so that the safety messages are verified in which messages from closer vehicles are given higher priority and farthest vehicle's messages are given lower priority. The solutions to achieve this can be categorized into two approaches. Vehicle status based and channel-aware based prioritization approaches. The former approaches completely rely on mobility information within BSM (i.e. speed, direction, acceleration, headings, etc), that will leads the schemes to be vulnerable to different security threats like broadcast tampering and denial of service (DoS). On the other hand, the latter used the received signal strength of BSM to classify the messages into five fixed safety areas, so that the security issues in the former approach covered. However, still, the messages within each safety area need to be prioritized to satisfy the demands of ITS application which recommend that nearby vehicle's BSM need to get verification time before far vehicles even within their corresponding safety areas.

In this paper, we proposed Enhancing Intra Safety Area Queuing (ISAQ) for a Channel-aware-based BSM verification scheme. We design safety messages prioritization scheme by using the hybrid of the two existing approaches. To classify the messages into different safety areas, we used the received power of the message. And to rank the messages within their safety area, we used the combination of transmitter-receiver distance and messages arrival time at the vehicle's buffer. From the simulation result, we observed that our modified safety message prioritization scheme for verification is more secured during BSM classification and efficient during BSM ranking than existing schemes.

**Keyword:** *VANETs, K-means clustering, safety areas, prioritization, verification, basic safety message*

# Chapter One

# Introduction

## 1.1. Background

In recent years, the transportation industry's rapid growth makes it easy for people to transport goods around and travel in a short duration. The main challenge in transportation is how to improve road safety. The report in 2015 by World Health Organization (WHO) shows that the road traffic death number globally has reached 1.25 million per year and injured people are 20 to 50 million [1]. According to [2], 94% of accidents occurred because of poor or wrong decision making of drivers. In general, 80% of drivers did not pay attention within a few seconds of an accident.

Experts in the transportation industry have been searching for services to increase safety and provide information among vehicles. To accomplish this goal, Intelligent Transportation Systems (ITS) have proposed to exchange information among vehicles and infrastructure. This kind of network is known as Vehicular Ad hoc Networks (VANETs). Many researchers and companies in different countries like the US and Europe are trying to address challenges in the VANETs environment. In the US and EU, the results of projects are mainly used for standardization bodies in ITS. In the US, the research mainly focused on the protocol suite IEEE 1609 which enables vehicles to communicate wirelessly. In the EU, they are contributing to European Telecommunication Standards Institute (ETSI) ITS, and International Organization for Standardization (ISO) CALM (Continuous Air-interface Long and Medium range) standardization [3].

In VANETs, each vehicle is equipped with an On-Board-Unit (OBU) for storing and processing important information and necessary sensors like Global Positioning System (GPS). There are also stationary Road Side Units (RSUs) along roads. Vehicles can communicate directly with or indirectly with other vehicles and RSU to create a large-scale network for sharing necessary information. The requirement to share messages about traffic conditions is that each receiving vehicle/RSU needs to verify the messages to identify whether it's from valid or invalid sources. The approach is as follows: transmitting devices cryptographically sign and optionally encrypt messages while receiving devices verify the signatures and optionally decrypt the messages [4] [5] [6] [7].

Wireless Access in Vehicular Environment (WAVE) is one of the standards that allow vehicles and the infrastructure to communicate and share information wirelessly [5]. To improve traffic efficiency and to increase safety and awareness in a vehicle about its neighboring vehicles, according to WAVE suggestions vehicles should broadcast Basic Safety Messages (BSMs) which contain vehicles' status (such as position, velocity, and heading) every 100ms or 300ms to the one-hop communication range. After the received BSMs are verified, safety applications [4–7] use the information in BSMs to increase traffic efficiency and prevent vehicles from possible safety incidents by sending warnings to the drivers. In highly dense VANETs, a vehicle may receive thousands of BSMs from neighboring vehicles. Due to the message verification process involves a time-consuming cryptographic operation, it makes it impossible for a vehicle to verify all messages [8]. According to [9], it could take at least 4.97ms on average for one message to be verified. So, the BSM receiving rate is usually higher than the verification rate. To cope with the problem, a vehicle can either accept BSMs without verification or verify as many BSMs as it can and discard the rest. The first method can cause security issues where the vehicle can be exposed by the attacker, and on the other hand, the second method can impact safety application [15] to miss relevant and important information. So the best approach is developing an algorithm for a vehicle to selectively verify messages based on their potential relevance to the safety applications.

Different papers have been proposed on receiver side safety message prioritization schemes to verify the message based on different approaches. In [8] [10] [11] [12] and [13] currently, there are two main safety messages approaches vehicle status based prioritization schemes and channel-aware based prioritization scheme. Existing vehicle status-based prioritization schemes completely rely on mobility information within BSM (i.e. speed, direction, acceleration, headings, etc), which will lead the schemes to be vulnerable to different security threats like broadcast tampering and denial of service (DoS) [10]. Therefore, it can impact the safety of the end-to-end ITS application, when undeserved priority can be served during zone creation. The channel-aware-based prioritization scheme used the received signal strength of BSM to cluster the messages into five fixed safety areas so that the security issues are covered. However, still, the messages within each safety area need to be prioritized to satisfy the demands of ITS application which recommend that nearby vehicle's BSM need to get verification time before far vehicles even within their corresponding safety areas.

To solve the specified problem of the existing prioritization scheme, we proposed a novel Enhanced Intra Safety Area Queuing (ISAQ) for a Channel-aware-based BSM verification scheme in VANETs.

## 1.2. Statement of Problem

WAVE standard suggests that every vehicle should periodically broadcast BSMs to its one-hop neighbors at an interval of 100ms or 300ms. But, this could lead to a large number of safety messages to be verified [14] [15]. Elliptic-Curve-based Digital Signature (ECDSA) is time consuming cryptographic operations, which was measured to take 4.97ms on average to verify a single BSM. For instance, in dense highway VANET with 100ms broadcast interval and assuming 100 vehicles in its one-hop receiving range (which is reasonable for a multi-lane road), a receiving vehicle may receive 1,000 safety messages per second. But, the verification rate is only 200 messages per second, which is much lower than the received messages (i.e. 1000 messages). Therefore, addressing the mismatch between the BSM arrival rate and the verification rate problem is very important. Many papers have been proposed to address this issue with two common schemes.

The first scheme in [8] [11] [12] and [13] is vehicle status based prioritization scheme, which completely relies on the BSMs mobility information to classify the road into different safety areas. Since the mobility information in BSM can be exposed before verification, complete reliance of existing schemes [8-14] on this information can result in the schemes be vulnerable to different security threats, like broadcast tampering and denial of service (DoS) [9]. Due to this, undeserved priority can be served especially during zone creation. Thus, it can impact the safety of the end-to-end ITS application.

The second scheme in [10] is a channel-aware-based prioritization scheme. It utilized the received signal strength of BSM to cluster the messages into five fixed different road safety areas (SA). Then the BSM assigned to the highest safety area (SA1) will get verified before BSM within the lowest safety area (SA5). The approach reduced the security problem in existing schemes [8-14], as the scheme gets rid of using mobility information in BSM in process of prioritizing the messages for verification. However, its key limitation is the way the BSMs within their safety areas are extracted from multi-level priority queues for verification. The problem of using the scheduling technique to extract the BSM for verification is that it does not allow the receiving vehicle to consider the status of its neighboring vehicles.

To address the problem in the [10] scheme, we proposed a novel Enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in VANETs. Our proposed solution, rank the incoming safety messages within their corresponding safety areas using two parameters (i.e. transmitter-receiver distance and BSM arrival time). So nearby vehicle's BSM always get verification time before other farthest. We have used here also arrival time if two transmitting vehicles have a similar distance from the receiving vehicle. Therefore, in our proposed work, the queuing delay of nearby BSM always low and awareness accuracy is improved and the number of BSM verified with transmitter and receiver distance is high compared to the existing technique.

The motivation behind to propose our new scheme is by considering the advantages of the two schemes. That is, we classified the highway road into different safety areas using BSM received signal strength (i.e. second scheme), so less vulnerable to security threats and then, we ranked BSM within their safety area using vehicle status (i.e. first scheme), so the receiver is aware of transmitting vehicle's status during ranking the BSMs. As a result, the modified safety message prioritization scheme for verification will be secured and efficient than the existing one.

Generally, the following questions are answered in this thesis:

- How to rank safety messages within their corresponding safety area?
- How to extract nearby vehicle's safety messages for verification with low waiting time?
- How to improve awareness accuracy between neighboring vehicles via ranking BSMs using transmitter-receiver distance and BSM arrival time?

## 1.3. Objectives of the Research

### 1.3.1. General Objective

The general objective of this paper is to develop Enhancing Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme on highway VANETs.

### 1.3.2. Specific Objective

The following specific objectives are set to accomplish the general objective:

- Investigate the current safety message prioritization schemes on highway scenarios.
- Design Enhancing Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme on highway VANETs scenario.
- Implement the proposed solution on simulator environment on highway scenario.
- Test and evaluate the performance of the proposed solution via simulations to prove that the proposed work enhanced the existing scheme.
- Compare and contrast the new scheme with existing schemes.

## 1.4. Scope and Limitation of the study

The scope of this thesis is limited to designing and implementing Enhancing Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in highway VANETs scenario. The approach contains an algorithm for BSM ranking within their corresponding safety areas. The proposed solution will optimize the performance of the existing [10] scheme by ranking the safety messages using transmitter-receiver distance plus BSM's arrival time to extract the message from the MLPQ module for verification. So the waiting time of nearby vehicle's BSM in security queue to get verification time enhanced and cooperative awareness accuracy between neighboring vehicles also improved when compared with the existing scheme.

The thesis will not cover the following issues:

- Routing issues.
- Malicious vehicles detection that denies verifying received message issues
- Isolating vehicles of invalid BSM from the network.

## 1.5. Methodology

### 1.5.1. Literature Review

To accomplish the objectives of this thesis, several resources that related to the work are investigated, such as published international journals, conferences, articles, books, workshops, related websites, and other vital documents to fully understand the VANET system and existing receiver side safety message prioritization schemes.

### 1.5.2. Design and Implementation

In the design phase, proposed solutions on highway dense scenarios which are specified in the objectives of this thesis are will be designed. Due to the prohibitive costs of employing VANETs, different wireless access technologies, and vehicles in real-world testbeds, we will implement the proposed solution using a simulation VANETs environment.

### 1.5.3. Evaluation of the Proposed Work

The experiment will be conducted to test the usefulness of the proposed schemes on highway dense scenarios and will be evaluated in terms of their objective and contribution in comparison to what is already done in the simulation environment.

## 1.6. Significance of the study

VANETs have a lot of potentials for many applications to be developed for ITS. Different types of data are monitored with VANETs applications. For instance, vehicle conditions, surrounding roads, neighboring vehicles, the surface of the road, and weather. The data is available for different purposes [13]. Based on their purposes, the VANETs applications can be divided into non-safety applications and Safety applications. Non-safety applications provide comfort for road travelers and also make the journey more enjoyable. Some examples are Infotainment, Payment Services, and Traffic/route optimization. Safety applications have the focus on decreasing the probability of traffic accidents and loss of life [14]. Some of the road safety applications which use V2V communication are cooperative forward collision warning, lane change warning, blind-spot warning, and visibility enhancement applications.

Hence, the contribution of this work will improve the BSM waiting time in the security queue to get verification time and also enhance the awareness accuracy between neighboring vehicles, through ranking BSM within their corresponding safety areas, so that satisfy demands of

VANETs safety applications which suggests nearby vehicle's safety messages need to get verification time so that the possibility of the accident to occur will be reduced.

## 1.7. Thesis organization

This thesis is organized as follows. Chapter 2 presents literature about the nature of VANETs and different wireless access network technologies. Chapter 3 introduces related works that are carried out for improvement safety message prioritization techniques. Chapter 4 presents the detail of the proposed work with its architecture and algorithms. Chapter 5 provides a simulation study and evaluation of the proposed algorithms. Finally, Chapter 6 presents the conclusion, contribution, and future works.

# Chapter Two

# Literature Review

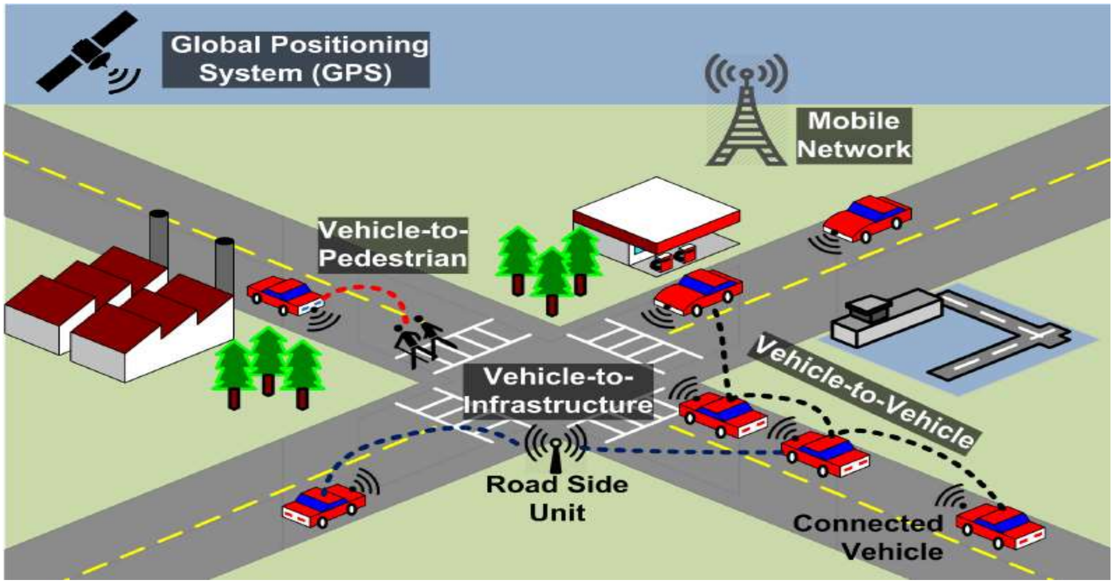
## 2.1. Overview

VANET (Vehicular Ad-hoc Network) [16] is a type of network that connects an enormous field of mobile circulated applications that runs in the vehicle. VANET is a sub-part of the Mobile Ad-hoc Network (MANET) where the vehicle acts as the mobile nodes within the network picture with a stay on a connection. The nodes are vehicle and roadside units and they can communicate with one another via one hop (directly) or multi-hop (indirectly). VANET is a key component of the Intelligent Transportation System (ITS) which had better dispensation and storage capability. The main goal of VANET is to provide road safety information among the nodes so that the frequent exchange of such type of data on the network signifies the role of safety.

The Intelligent Transportation Systems (ITS) main aim is to provide a solution for protecting passengers from possible accidents and traffic congestion problems. The ITS has improved the driving environment by integrating information technology in transport systems. The most common types of communications architecture of VANETs are:

- **Vehicle-to-Vehicle (V2V):** Its communications architecture provides interaction within vehicles. In V2V, a vehicle can broadcast and exchange traffic conditions with other vehicles.
- **Vehicle-to-Pedestrian (V2P):** Its communication type in which vehicles share and communicate important information with passengers. It provides the connection between the vehicle and roadside users using the V2V application.
- **Vehicle-to-Infrastructure (V2I):** Its communication type in which the information will be broadcast between the nodes (i.e. vehicle) and the infrastructure (said as ITS), to deal with important information such as road conditions and safety events that have been taken into account. In this V2I, a vehicle (node) launches a connection between RSU and contact with external networks which is the internet [17]. Figure 2.1 shows the architecture of communication in VANETs.

- **Vehicle-to-Sensors (V2S):** It is a communication type that provides the communication between sensors in intra-vehicle sub-network;




[17]

*Figure 2.1 Communication Architecture in VANETs*

## 2.2. Characteristics of VANETs

VANET is a network without the infrastructure in which the node can be either the moving vehicle or Road Side Unit (RSU). It provides the characteristics of ad hoc network and wireless medium methods that use a different topology for communication and infrastructure dependent modes. VANET is a sub-part of MANET application which had its distinct characteristics [18] that can be explained as follows:

- **High mobility:** In VANET, the node moves at a high speed that condenses the mesh in the network [9] [10].
- **Rapidly changing network topology:** The node in VANET is highly mobile and the speed of the vehicle should also random so that the node position will change frequently. The topology is dynamic and unpredictable. It facilitates the entire network attacks and makes it hard to find misbehavior in the network [10].
- **Availability of the transmission medium:** VANET can be implemented for one city, several cities, or countries. This means that network size in VANET is geographically unbounded. The universal availability of this wireless transmission medium is great

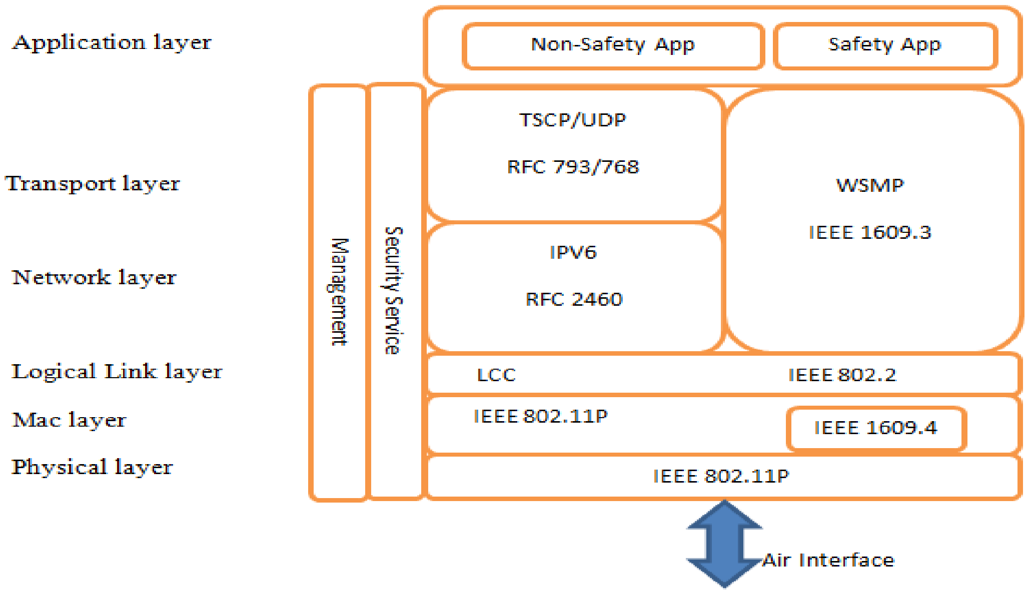
advantages in Inter-Vehicle Communication (IVC), becomes the origin of some security issues, related to both the nature of transmission in a wireless environment and to the security of communications using open support [10][11]

- **Frequent exchange of information:** Normally the VANET is ad-hoc in nature. It inspires the nodes to collect information from the neighbor vehicles and roadside units. So that, the nodes exchange their information periodically [9] [10][11].
- **Limited bandwidth:** In VANET, the standard DSRC band should be measured as limited, the width of the DSRC band was 27 MHZ. The throughput was 27 Mbps which is a theoretical value [10].
- **Energy storage and computing:** The VANET nodes have no issue of energy, computing capacity, or storage failure. This allows VANET usage of demanding technique such as RSA, ECDSA implementation and also provides unlimited transmission power [10] [11].
- **Time-critical:** Within the time, the information in VANET should send to the accurate node, so that the node will make a decision and execute action correspondingly [10].
- **Better Physical Protection:** In VANET the vehicle should be well protected physically. Therefore, physically compromising the VANET node will be difficult and it is very difficult to reduce the outcome of infrastructure attack [9] [10].
- **Limited transmission power:** In the WAVE the transmission power should provide up to the data reached. The data reachability distance can be said to be 1000m. For crisis and any public safety such as accident problem or any traffic congestion problem, it is allowed to transmit with a high power [8][9]
- **Variable Network Density:** This depends on the density of traffic, which can be low, as in suburban traffic, or high during traffic jams [10] [11].
- **Services of safe driving:** This is achieved by improving traveler satisfaction and improving traffic efficiency. The direct communications between mobile nodes are ensured by VANETs, hence enabling the usage of a set of applications that require direct communication between vehicles over the network. These applications offer warning information to passengers moving in the same direction concerning the urgency for swift hard breaking or about accidents, thus the driver needs to create a larger image of road topology ahead. Furthermore, VANETs can also improve traveler satisfaction and improve traffic efficiency by providing information such as shopping malls, gas stations, weather, traffic flow, and fast food [8][9][10].

## 2.3. Communication Technology in VANET

Federal Communication Commission (FCC) in US government agency is established for regulation and licensing for 75 MHz spectrum ranges from 5.850 to 5.925 GHz band which is known as Dedicated Short-Range Communications (DSRC) service in ITS. The 75 MHz spectrum defined in [19] [20], is divided into 7 channels of 10 MHz and 5 MHz guard band. One of the well-known communication standards in ITS is WAVE [5]. WAVE makes it possible for vehicles and infrastructures to communicate with each other. The protocol stack that supports the application layer is comprised of the WAVE 1609 standards family [5] [21]. The main reason behind developing this type of standard is to increase safety on road by making possible communication between vehicles and infrastructures.

WAVE standard uses the DSRC frequency band for exchanging information between entities in the VANETs [22]. Since WAVE can offer low latency, wireless communication for safety applications makes it suitable to perform in a dynamic environment. In VANETs, the devices which use WAVE can have two categories: Onboard Units (OBUs) and Road-side Units (RSUs). OBUs are used in mobile stations (vehicles) and RSUs are used as base stations. OBUs and RSUs can communicate directly or indirectly with each other in VANETs. In our thesis, we assume that the architectural components of a VANET (such as On-Board Unit, Road Side Unit, and wireless interface) are compatible with the IEEE 1609 family of standards for WAVE [5]. Figure 2.2 shows the protocol stack for WAVE. The main standards in uses in the stack protocol of WAVE are summarized as follows: IEEE 1609.2 (for security services), IEEE 1609.3 (for network services including the WAVE Short Message Protocol), IEEE 1609.4 (for multi-channel operation), and IEEE 802.11p (for wireless MAC and PHY specifications.)

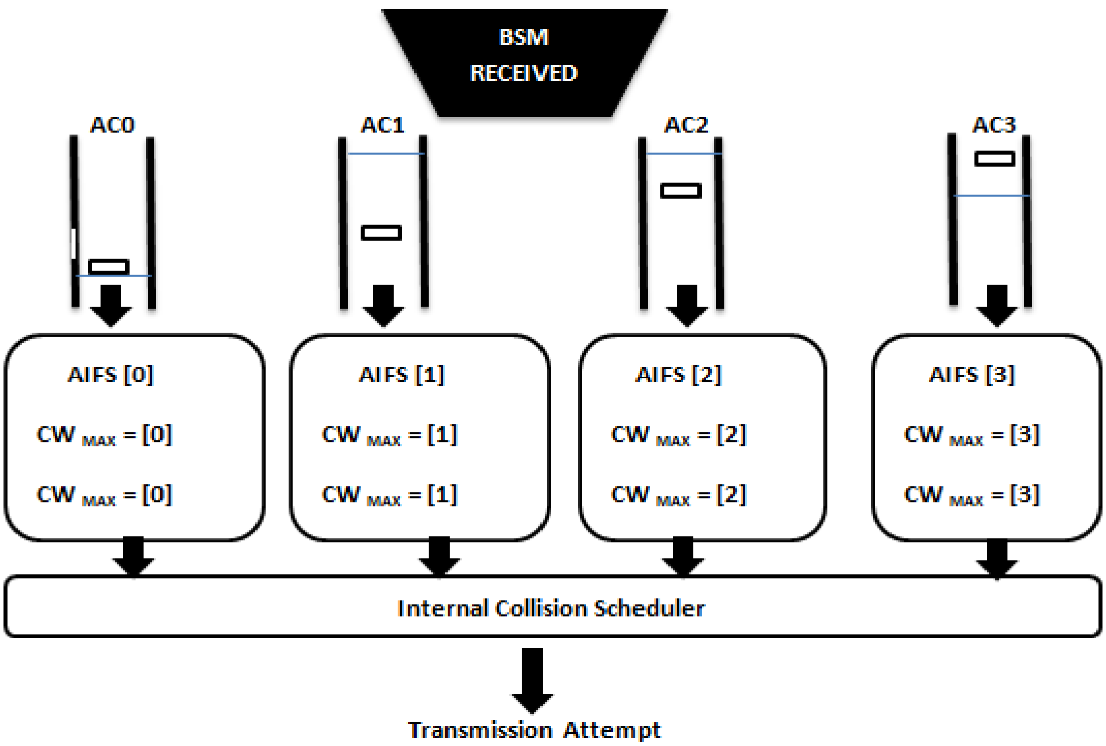



[12]

*Figure 2.2 WAVE standard stack protocol*

As shown in the above figure, WAVE uses 802.11p at the physical layer [21]. It's a modified version of the IEEE 802.11 standard that was divided into two sub-layers: Physical Medium Dependent (PMD) and Physical Layer Convergence Procedure (PLCP). The first one is to utilize the Orthogonal Frequency Division Multiplexing (OFDM) technique and the second one defines the mapping between the MAC frame and the basic physical layer data respectively [22]. IEEE 802.11p can transmit data at high rates from 3 to 27 Mbps over 10 MHz bandwidth. It has the aim of providing communication between vehicles and infrastructures up to 1000 meters.

In WAVE, the Data Link layer has two sub-layers: Medium Access Control (MAC), and Logical Link Control (LLC). The MAC defines how to access a common medium. MAC layer uses IEEE 802.11e to provide quality of service [23]. IEEE 802.11e uses Enhanced Distribution Channel Access (EDCA) mechanism to give priority to more important services. EDCA is comprised of four separate FIFO buffers called Access Categories (AC) from AC0 to AC3 where AC0 has the highest priority. Therefore, AC0 has an access to the channel more compared to other ACs. Figure 2.3 shows the four different transmit buffers for each AC. Each AC buffer has a different Contention Window (CW) size and Arbitration Inter-Frame Spacing (AIFS). The smaller AIFS value for AC gives higher priority to access to the channel for transmitting the data [24].

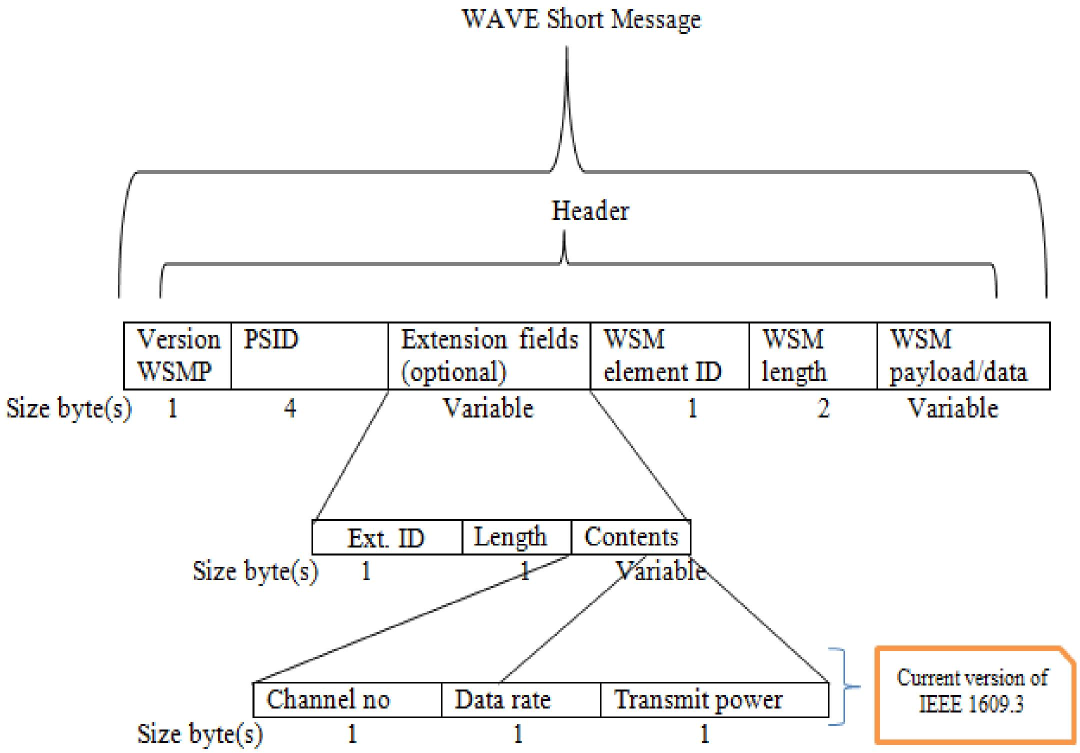



[13]

*Figure 2.3 ECDA prioritization mechanisms in WAVE.*

The Logical Link Control (LLC) uses IEEE 802.2 in cooperating with the Sub-network Access Protocol (SNAP) [25] [26] to support IEEE 1609.3 [27]. They require no-acknowledgement connectionless service with Unnumbered Information (UI) frames. In WAVE, the protocol associated with LLC payload is specified by Ether-Type which has the two known values are 0x88DC (WAVE Short Message Protocol) and 0x86DD (IPv6).

In WAVE, the network and transport layers are found above the LLC layer. They are classified into IP-based and non-IP-based data transmission. The non-IP based data transmission uses IEEE 1609.3 standard [27] to define and transmit WAVE Short Message (WSM) via WAVE Short Message Protocol (WSMP) were primarily meant for safety applications in VANETs [22] [28]. The IP based data transfer uses traditional internet protocols, IPv6, UDP, and TCP. Generally, the services depend on their requirements by choosing to use either WSMP or IPv6 service for transmission of data. Typically, the overhead of the WSMP packet is 11 bytes less than UDP/IPv6 packets which have a minimum size of 52 bytes [29]. Figure 2.4 shows the format of WSM consisting of headers' size and payload.




*[12]*

*Figure 2.4 WSM packet format.*

As Figure 2.4 above showed that the header of WSM is 1 byte which indicates the WSM version. WSM uses Provider Service ID (PSID) field with 4 bytes size to identify the applications. It has a similar function with TCP/UDP packet's port number. The Extension field is an optional field of 3 bytes size, which is used for flexibility in communication. WSM element ID indicates payload format and shows the end of Extension fields. WSM length field shows the size of payload which has 2 bytes size. WSM payload field contains information that comes from the application layer. In general, an On-Board-Unit (OBU) uses a First-In-First-Out (FIFO) buffer, at the network and transport layer, to handle receiving messages from entities in a VANET.

Finally, applications can be classified into two categories at the application layer: non-safety and safety. Non-safety applications refer to those which are used for infotainments and advertisements. Safety applications refer to applications that are used to detect and prevent vehicles from having incidents such as accidents. A common message set used by safety applications is defined by SAE J2735 [28].

## 2.4. Applications of VANETs

VANETs have a lot of potentials to develop many ITS applications. Protocol stack of VANETs provides applications requirement for vehicular environments and different types of data can be monitored using applications of VANETs. For instance, traffic conditions, surrounding roads, neighboring vehicles, and weather. The data is available for different aims. The vehicle communicates with its neighboring vehicles to exchange the relevant information [30]. Based on their purposes, the VANETs applications can be divided broadly into two categories. These are non-safety applications and safety applications.

### 2.4.1. Non-Safety Application:

Non-safety applications refer to applications that provide comfort for road travelers and also make the journey more pleasant. In case of comfort for road traveling, it can refer to traffic efficiency and management applications to improve traffic flow, traffic coordination, and traffic assistance such as speed management, and co-operative navigations applications [31]. The other applications related to infotainment can be local services or global services. The local services which focus on local-based services are the following: Point of interest advertisements, Maps download, Parking payment, and automatic tolling services. The global services which mainly focus on data that can be obtained from the Internet are the following: Insurance, Parking zone management, financial services, web browsing, and Voice over IP [3] [31].

### 2.4.2. Safety Application:

Safety applications are applications that have the main focus to decrease the probability of traffic accidents and loss of life [31] [32] [33]. A significant number of accidents happening in the world every year related to intersections, bind-spot, rear-end and lane change collisions. Safety applications use information collected by vehicles from its neighboring vehicles to alert a driver to prevent such collisions with other vehicles. Some examples of safety applications are as follows**:**

- **Blindspot warning application:** This application is designed to alert a driver when there is a vehicle at the blind spot when a vehicle wants to change lanes.
- **Visibility enhancement application:** This application is used for alerting a driver when there is an unsafe situation occurring when there is low visibility due to heavy rain, fog, storm, or others.

- **Cooperative forward-collision warning application:** It is an application which designed to alert a driver when there is a potential of rear-end collision to vehicle ahead. In general, the application uses position, velocity, acceleration, heading, and yaw-rate to analyze the unsafe situation.
- **Lane change warning application:** This application is used to alert a driver when there is a potential collision for changing lanes. When a driver wants to change lanes and uses a signal for changing lanes, the vehicle uses the information of other vehicles such as position, velocity, acceleration, and heading to analyze the situation such as calculating the gap between vehicles for safe lane changing.
- **Highway merging assistance application:** Alerts a driver when a vehicle at the blind spot or a vehicle is on a highway ramp trying to merge. The vehicle uses the heading, position speed of that vehicles to analyze the situation and alert a driver if there is an unsafe situation. In Table 2.1 we have summarized the requirements of safety application such as transmission mode, allowable latency, and the maximum range.
- **Cooperative collision warning application:** It alerts a driver when there is a potential accident about to happen. The application uses the collected information of neighboring vehicles such as position, speed, acceleration, wheel angle to analyze them with its sensor information for a potential collision.
- **Pre-crash sensing application:** Far way vehicle becomes active when there is an accident about to happen with a vehicle. This application uses neighboring vehicle information to detect this kind of situation.

*Table 2-1. Safety applications and their specific requirements.*

| Application | Transmission Mode | Allowable Latency (s) | Maximum range (m) |
|---|---|---|---|
| Cooperative forward-collision warning | Periodic | 100 | 150 |
| Lane change warning | Periodic | 100 | 150 |
| Blind spot warning | Periodic | 100 | 150 |
| Highway merge assistance | Periodic | 100 | 250 |
| Visibility enhancement | Periodic | 100 | 300 |
| Cooperative collision warning | Periodic | 100 | 150 |
| Pre-crash sensing | Event-driven | 20 | 50 |

[31]

## 2.5. Types of Message used for Safety Applications in VANET

SAE J2735 over Dedicated Short Range Communication (DSRC) is a standard for messaging in VANETs [34] [35]. This standard defines fifteen types of messages used in VANET communications. Basic Safety Message (BSM) is an important message type used by vehicle-to-vehicle safety applications or cooperative safety driving applications. For the rest of this thesis, a BSM is referred to as a safety message or message. Each safety message has the default size of 254 bytes.

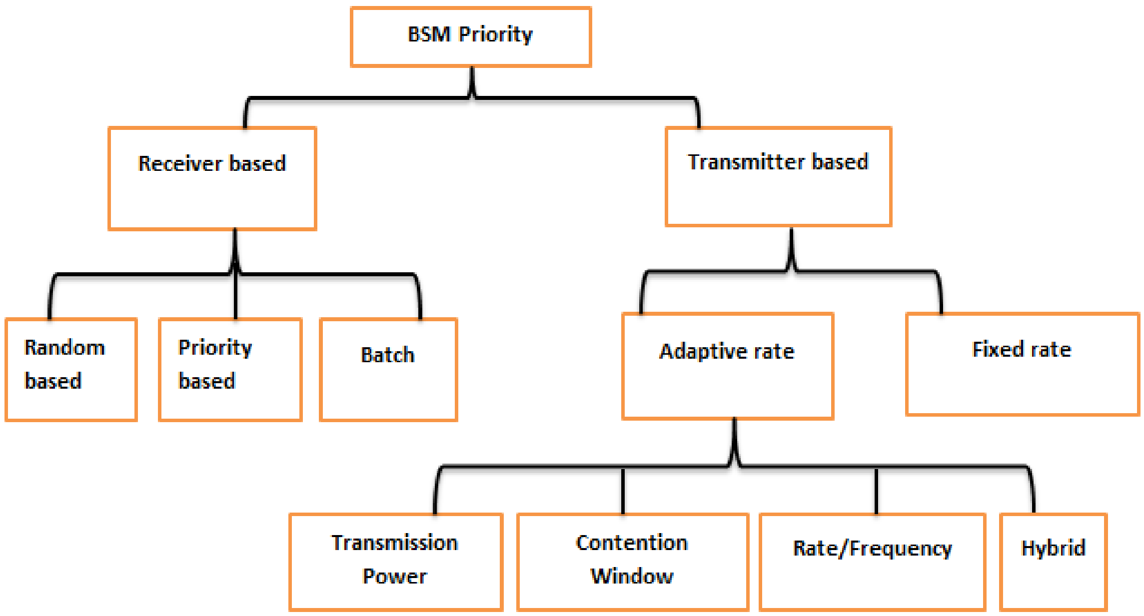
Vehicles usually broadcast safety messages to inform neighbors about their statuses at either 100 milliseconds or 300 milliseconds intervals. During safety message delivery, in order to avoid delay, there is no acknowledgment or handshaking. They are broadcast to all one-hop neighbors. According to the WAVE standard, vehicles can communicate up to the range of 1km. The maximum communication range can be used by vehicles for specific purposes such as sending emergency messages or routing messages. The National Highway Traffic Safety Administration (NHTSA) recommends an operational range of up to 300 m for vehicle-to-vehicle communication [36]. Each safety message contains information about the status of a vehicle, such as velocity, direction, acceleration, and optional information such as event flags. Periodically broadcasting safety messages by all vehicles allows other vehicles to be aware of nearby vehicles.

Some safety applications are required to transmit messages periodically (for example, every 100 milliseconds), whereas other safety applications need message transmission when an event occurs [37]. Safety applications examine messages and provide essential action if needed to prevent or warn a driver from about to happen situation. Thus, in general, safety messages can be categorized into two groups. These are periodic messages (messages that are transmitted for awareness of the environment) and the other messages are event-driving messages (event messages which are triggered by unsafe situations).

- **Periodic Messages:** This is an important type of message in safety applications. It is also known as Basic Safety Message (BSM) for V2V communications. Generated BSM is used for neighboring vehicles to have a clear and accurate awareness of potential threats/crashes 360 degrees around the vehicle. Vehicles notify the neighboring vehicles about their existence by transmitting this message. This message contains necessary sensor information of vehicles such as the speed of the vehicles, acceleration, heading, wheel angle. Usually, periodic messages are broadcast in a range of 300 meters radius around a one-hope distance of the vehicle. A vehicle can prevent an unsafe situation by processing these messages before it happens.
- **Event driving messages:** This type of message is also known as emergency messages. They are transmitted to neighboring vehicles if an incident/unsafe situation has been discovered. Thus, this type of message will not be generated, if there is no incident occurred. Event message has the highest priority for a vehicle to process and usually, it contains location, time, and event type.

## 2.6. Safety Messages Prioritization schemes in VANETs

According to [13] [14], safety message prioritization approaches can be categorized into transmitter side and receiver side safety message prioritization schemes for verification. Given Figure 2.5 below shows the classification of the two approaches.




[13]

*Figure 2.5 BSM prioritization schemes categories*

Let us discuss the advantage and limitations of these two approaches one by one.

### 2.6.1. Transmitter-side Safety Messages Prioritization schemes

Safety message prioritization at the transmitter-side is performed based on transmission rate, transmission power, contention window size, or a combination of the aforementioned factors.

- **Fix rate transmission of safety messages:** WAVE standard utilize fixed rate to transmit safety messages in VANET (i.e. 10 message/s) [32]. The WAVE protocol [5] provides quality of service at the MAC layer by adopting the Enhanced Distributed Channel Access (ECDA) with four separate buffers (AC3, AC2, AC1, and AC0, in descending order of priority) to prioritize transmitting messages. Messages in the buffer with higher priority (i.e. AC3) will have more chances to access the channel. In [40] proposed the Oldest Packet Drop (OPD) buffering mechanism at the transmitter to increase the freshness of the messages sent. OPD scheme is better than the prioritization scheme in the WAVE protocol, in which messages are transmitted in First Come First Serve fashion and new messages are dropped when the transmit buffer is full. However, with this scheme, in very dense traffic, the freshness of messages may be decreased at the receiver due to queuing and processing delay.

- **Adaptive rate transmission of safety messages:** This scheme adjusts the adaptive rate of safety messages transmission based on the condition of the VANET. Paper [41] proposed to use of clustering vehicles based on their mobility. Each cluster is assigned a cluster head based on its relative speed and distance to cluster members. The cluster head determines the data propagation inside and between clusters. In [42] proposed dynamically adjusting of the beacon transmission rate based on current traffic density, while maintaining appropriate accuracy to increase the performance of VANETs in a high-density traffic condition. The proposed mechanism uses the movement of neighboring vehicles such as velocity and acceleration to estimate the transmission rate of a beacon. However, the drawback of these schemes is that low rate transmission rates may cause inaccuracy in safety applications and reduce the awareness of the vehicle about the status of neighboring vehicles in the vicinity.
- **Safety messages' adaptive transmission power:** This scheme adaptively adjusts the range of communication by increasing or decreasing transmitting power. The higher the transmitting power the farther range, a vehicle can broadcast messages. As a result, the lower transmission power can give the closest vehicles higher priority. In [43] proposed a delay-bounded dynamic interactive power control algorithm in which each vehicle iteratively uses a directional antenna to adjust the transmission power for neighboring vehicles [44] focused on increasing the probability that neighboring vehicles receive beacon at the maximum possible range of communication. Their scheme uses piggyback over beacon to share the transmission power control information with neighboring vehicles. In [45] used network topology persistent scheme based on the density of network to adjust the transmission power with acceptable coverage percentage. However, the drawback of this scheme is reducing transmission power impact on the number of vehicles that can receive the message. This causes a significant reduction in the awareness of neighboring vehicles in the vicinity.
- **Adaptive contention window size for transmitting safety messages:** This scheme adaptively adjusts the contention window size (CW) of MAC in 802.11p WAVE protocol. As a result, reducing the CW parameter can give higher priority to the relevant messages for transmitting which causes the reduction in transmission delay for these messages. In contrast, increasing CW gives lower priority to irrelevant messages for

transmitting. In [46] proposed to adjust adaptively the parameter in the MAC layer such as CW and network layer to achieve the optimal value for VANET to transmit the message. In [47] used one-hop neighboring vehicles density and many vehicles that aware of them at time $t$ to estimate the value of CW. But, the drawback of this scheme is that increasing the CW harms transmission delay, and each time the transmission failed the value of CW will be doubled.

- **Hybrid adaptive transmitting safety messages:** This scheme uses a combination of adaptive transmission rate, power, and contention window size for transmitting BSM. In [48] [49] used traffic characteristics such as local vehicle density, traffic flow, and road segment size to determine the transmission range and then calculate the transmission power. The CW size is adaptively adjusted in EDCA to prioritize messages in the ACs' buffer.

### 2.6.2. Receiver-based Safety Message Prioritization Schemes

Even though safety message prioritization at transmitters can reduce the message arrival rate at receivers, it does not count the receiver capability and neighboring vehicles' messages. Thus prioritization of safety messages at a receiver is needed to verify more BSMs from transmitting vehicles in the vicinity which are more likely to be involved in a safety incident. The receiver-based prioritization scheme can be categorized into three schemes: random-based, batch, priority-based schemes.

- **Random Based verification Scheme:** To enhance the security and scalability of the system [7] proposed a verification scheme that chooses messages randomly from the buffer. Although this method has been used in several authentication schemes [8] due to its simplicity, the main drawback of this method is that some important messages may not get verification on time or not be verified at all.
- **Batch verification Scheme:** In this scheme, a receiver collects arrival BSMs as a batch and then verifies all at once [9]. So that, this verification scheme minimizes the verification time per BSM. The disadvantages of this scheme are: i) collecting messages in a batch causes an additional delay for verification and ii) if a single BSM in the batch has a false signature, the batch may not be successfully verified.

- **Priority Based Verification Scheme:** In this scheme, a vehicle uses mobility information such as velocity, heading, and direction in the BSMs received from neighboring vehicles to prioritize arrival BSMs in a buffer.

## 2.7. Broadcasting approaches in VANETs

As stated in the literature [30] [38], safety message broadcasting approaches in safety application depends on their broadcasting techniques: one-hop broadcasting and multi-hop broadcasting.

### 2.7.1. One-hop Broadcasting Approaches

Messages those periodically exchanged by neighbor vehicles and that are not forwarded to other vehicles used one-hop broadcasting technique. The IEEE 1609.4 standard based on the 802.11p amendment manages multichannel operations at the 5.9 GHz band. It divides the available band, specifically into seven channels of 10 MHz bandwidth. Particularly, there is a Control Channel, two channels used at the end of the frequency band, and four Service Channels ready for safety and non-safety applications [36]. One-hop safety messages which used this standard are generated periodically at the rate of 10 Hz to offer updated information about traffic conditions in VANETs.

Generally, one-hop based safety messages broadcasting approaches provide local information. Therefore, the requirement of additionally feasible aggregation algorithms in safety applications covering a wide area limits their functionality in such scenarios. These operations increase the computational overhead of the applications, which may delay the detection and notification of dangerous situations, thus making them unsuitable in many scenarios.

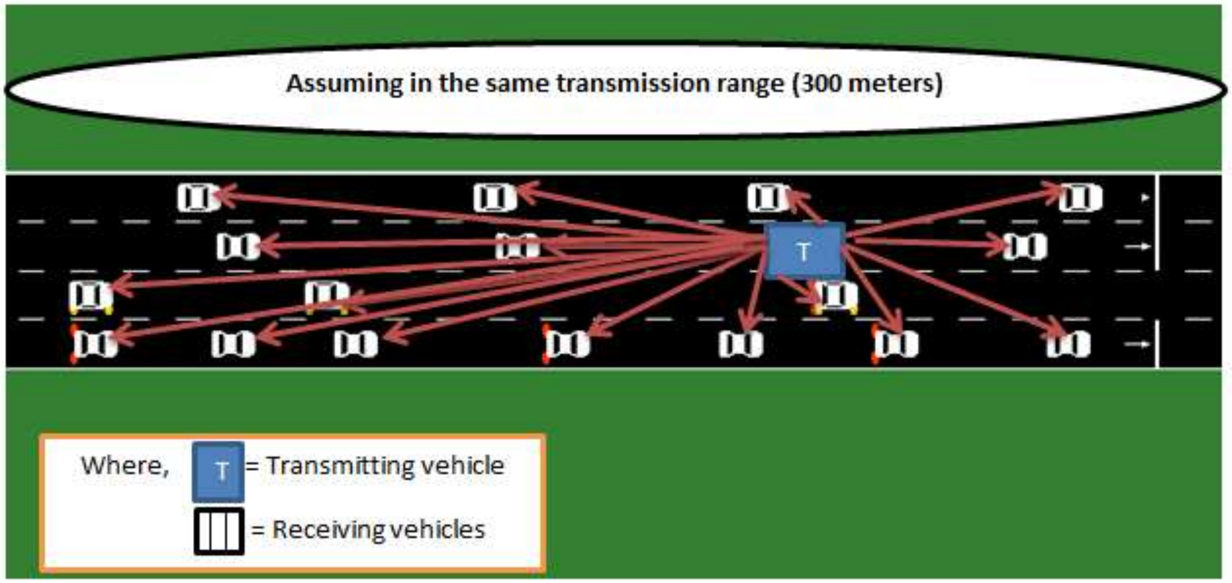



*Figure 2.6 One-hop data broadcasting approaches*

The above figure shows that, if vehicles need to exchange data (messages) about their status (i.e. speed, direction, acceleration, etc.), without additional intermediates vehicles, they can accomplish their communication.

### 2.7.2. Multi-hop Broadcasting Approaches

In this kind of broadcasting approaches, when an emergency is detected by a vehicle, the vehicle inform to its neighbor vehicle and the message should be rebroadcasted farther to notify the other vehicles that are not in the transmission range of the first vehicle [30] [37] [38]. Since VANETs are designed to support safety applications, the information is expected to be received by all the vehicles.

In VANET, safety message broadcasting is a critical issue to inform vehicles quickly about the accidents that may affect them. Different broadcasting approaches are designed to prevent broadcast storms by choosing certain vehicles from rebroadcasting using different parameters, hence contention in the channel, message redundancy, and collisions are reduced.

- **Flooding:** It is one of the data broadcasting approaches in which vehicles simply rebroadcast when they receive the message. Here if there are ‘k’ vehicles in the network, they simply rebroadcast for farther coverage of messages. When vehicles or RSU receive a message which has to be broadcast, initially they check whether the packet is new. If it is new, they rebroadcast; otherwise, they discard it. Since every vehicle forwards the message, it leads to redundancy. But, the message redundancy depends on the density of the vehicles found in the transmission range.

- **Safety Messages (Beacons):** Safety messages are messages that are periodically broadcasted by every vehicle to exchange information about their status (i.e. direction, speed, and other basic information). These messages have low priority than the alert (event) messages and they are broadcasted in one hope manner to the neighboring vehicles. They are not further rebroadcast by the neighbor nodes.
- **Store and Forward:** In this kind of broadcasting technique, when alert message received by a vehicle, the vehicle hold for some time until it gets other vehicles in its transmission range. According to this technique, a vehicle mostly waits to rebroadcast the message until a new neighbor is found. This way is mostly used in sparse network scenarios.
- **Probabilistic approach:** This technique depends on the probabilistic distributions to decide the probability of broadcasting the message, based on the conditions of the transmitting vehicle. Most of the broadcasting approaches that designed based on this technique use the Gaussian or the uniform distribution to associate a probability to each vehicle.
- **Distance-based approach:** According to this technique, the message rebroadcasting depends on the distance between the transmitting vehicle and the receiving vehicle. In this broadcasting technique rebroadcasting is not recommended if the distance between them is minimum, to cover large coverage.
- **Counter-based approach:** It is part of the flooding based data broadcasting technique. According to this, if (counter $c \geq$ threshold) for a received message, rebroadcasting then not allowed for that message. It is also known as limited flooding. Figure 2.5 demonstrates the multi-hop data broadcasting technique.

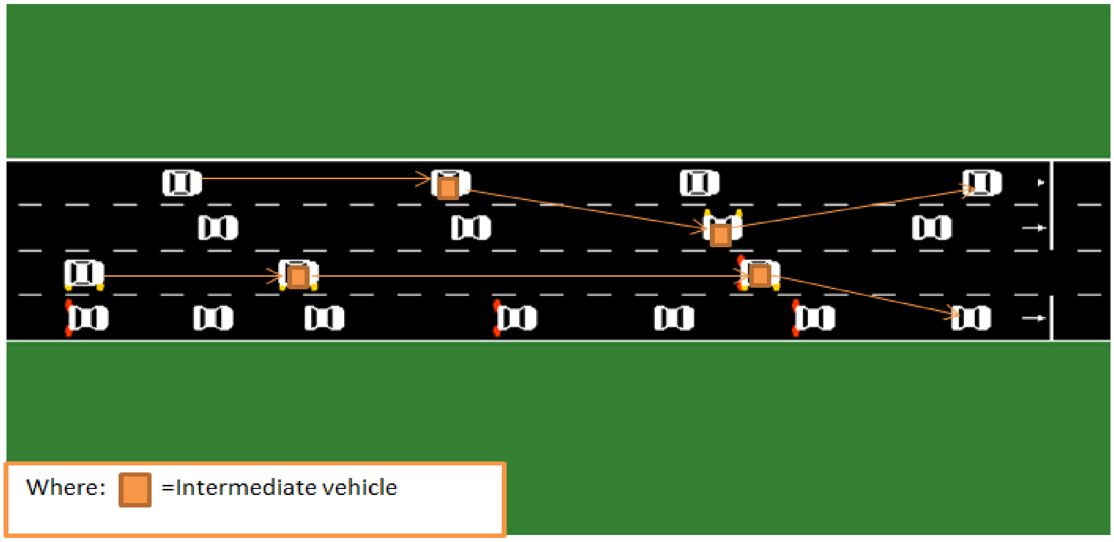



*Figure 2.7 Multi-hop data broadcasting approach*

In the above figure, the data is exchanged between source and destination through an intermediate vehicle which is orange colored.

## 2.8. Challenges in VANETs

As we expressed above, there are a lot of VANET applications in the ITS environment. However, to satisfy the demands of those applications effectively and efficiently, there are many challenges. The main requirements for VANET as explained in [16] [36], are packet loss reduction, bandwidth reservation, packet scheduling, and QoS control. Traditional approaches that are designed for MANET are not efficient and cannot be directly applied for VANET. According to the survey in [30] [39], the main challenges in VANET are the following.

- **Applications Heterogeneity:** VANET has various applications of safety and non-safety applications. These safety applications are time-sensitive that need low latency and high reliability while non-safety applications need low packet loss, better throughput and higher utilization of the resource. Therefore, designing an efficient and effective communication technique that can satisfy the demands of applications requirements is a critical issue in VANET.
- **Frequently Link Disconnections:** As it has been expressed above, vehicles have high mobility and travel at higher speeds (for example, over 100 km/hour) unlike nodes in

MANETs. This can result in the frequent change of network topology. Hence, there can be link failure from source to destination [30].

- **Disruptive Communications Tolerant:** At the moment there are problems such as lower reliability delivery and higher delay in low-density networks. To improve the delivery reliability, some solutions utilize the carry-and-forward technique, which in addition increases delivery time (i.e. high delay) of the information. Therefore, designing a mechanism without carry- and –forward stratagem is needed in VANET.
- **Protocols Standardization:** In VANETs, there can be different kinds of vehicles such as trucks, cars, taxis, motorbikes, bicycles and buses. In this kind of scenario, it's very indispensable that all of these vehicles can communicate with one another through the same protocol. Therefore, the challenging task here is creating a standard.
- **Broadcasting of Information:** Broadcasting emergency or alert information in VANET is a critical problem. The safety information in VANET requires broadcasting, unlike the other networks like the Internet, where data are typically unicasted [39]. Due to the fact that safety messages can be broadcasted to many of its neighboring vehicles instead of a single vehicle, to create awareness about an emergency situation, broadcasting those information using broadcasting technique is more comfortable than a routing approach which employs a unicasting approach. In the broadcasting technique, a vehicle does not require the address of the destination and the route to a particular destination. Broadcasting reduces a lot of difficulties in VANET such as route discovery, address resolution, and topology management complexity. Even though this approach is a better option, it can also cause the problem of blind storms in a dense network environment [36]. Therefore, designing a broadcasting technique that is capable of solving those problems is a challenging task.
- **Security Threats:** VANETs may face many challenges in the field of communication security and also in a revolution for vehicular safety and comfort in road transport. In the aforementioned applications, messages can influence driver behavior and consequently on road safety. Additionally, they can have economic consequences. Therefore, during the deployment of VANET, it is mandatory to consider the possible existence of adversaries or attackers who try to exploit the different situations. For instance, injecting false, modifying or repeating messages, and also impersonating vehicles. Therefore, the

security of communications in VANETs is an essential factor in preventing all these threats. Specially, in cooperative driving or awareness applications, where each vehicle transmits messages periodically (i.e. in the interval of 100 milliseconds or 300 milliseconds), the validation of the source of the received messages must be verified instead of accepting it as it is [13].

- **Safety message arrival-to-verification rate:** Safety message processing or verification plays a significant role in securing VANETs. As safety messages are broadcasted several times per second in a highly dense network, the message arrival rate can easily exceed the verification rate of safety messages at a vehicle. Therefore, designing an algorithm for selecting and prioritizing relevant messages from received messages to increase the awareness of vehicles in the vicinity is needed [12].

# Chapter Three

# Related Work

As we have discussed in Chapter 2, there are a two common ways of safety message prioritizing schemes for verification in VANET. These are, transmitter-side and receiver-side safety message prioritizing schemes. Since this thesis follows the receiver-based schemes, in this Chapter, we will discuss the existing receiver-side safety message prioritizing schemes. The receiver-based prioritization scheme can be categorized into three schemes. These are random-based verification, batch verification, and priority-based schemes.

*Random Based verification Scheme:* To enhance the security and scalability of the system [7] proposed a verification scheme that chooses messages randomly from the buffer. Although this method has been used in several authentication schemes [8] due to its simplicity, the main drawback of this method is that some important messages may not get verification on time or not be verified at all.

*Batch verification Scheme:* In this scheme, a receiver collects arrival BSMs as a batch and then verifies all at once [9]. So that, this verification scheme minimizes the verification time per BSM. The disadvantages of this scheme are: i) collecting messages in a batch causes an additional delay for verification and ii) if a single BSM in the batch has a false signature, the batch may not be successfully verified.

*Priority Based Verification Scheme:* There are two common approaches of priority based verification scheme. Vehicle status based and channel aware based prioritization schemes.

## 3.1. Vehicles Status based Safety Messages Prioritization Schemes

In this scheme, the receiving vehicle uses information in the BSMs (i.e. velocity, position, direction, and heading) received from neighboring vehicles to prioritize arrival BSMs in a buffer. Different papers have proposed this.

In [8] proposed a probabilistic verification scheme. Message ranking will be passed through three bloom filters. All bloom filter checks an assigned portion of safety messages against the existing entries in them. The outcome of the bloom filters is used in a binary decision tree to achieve the final rank k. Then, messages related to each rank will be randomly processed with a particular verification probability. However, verification based on probability may cause some received BSMs even from nearby transmitting vehicles to be not verified.

In [12] proposed the scheme that prioritizes BSMs based on location and direction of the transmitting vehicle (quadrant), close proximity (zones), and relative time. The key design of RTZ uses adaptive discrete zones based on human reaction time and density of network where the received messages from the close zone with lower relative time have higher chance to be verified.

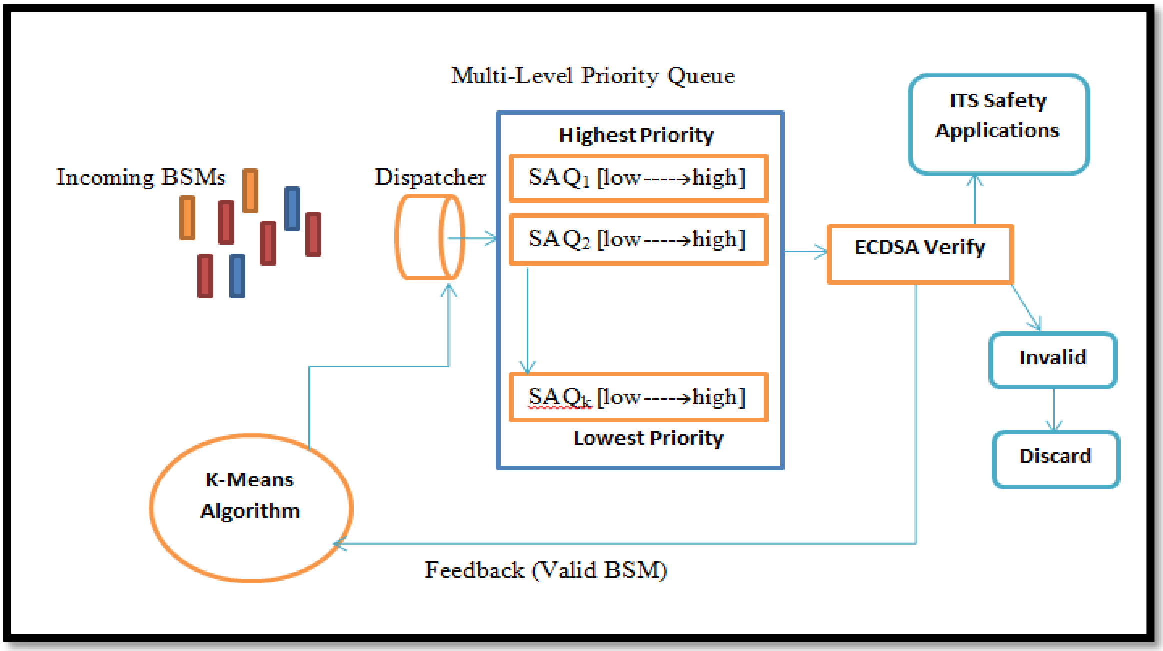
In [13] proprosed the an enhancement of RTZ [12]. The key design of the enhanced RTZ is they added history of BSM to be stored to avoid duplication of message verification. So that only the most up-to-date message from each vehicle is kept in the receiver's buffer.

However, approaches in [12][13] allowed to cluster and prioritize arrival BSMs into safety areas based on velocity, direction, and location of the transmitting vehicle from the receiving vehicle. This shows their scheme completely relied on vehicle status or mobility iformation in BSM. Thus, the absolute dependence on BSM clustering into different safety areas will lead their approach to be vulnerable to security threats since the integrity of this information is not yet verified.

In general, the key limitation of above schemes [8] [12] [13] are they all completely rely on the mobility information which can be taken advantage of before verification, which can result in all the approaches to be vulnerable to different security threats, such as broadcast tampering attack and denial of service attack [14], thus it will impact the safety of the end-to-end ITS application and also result in the prioritizing approach to be inconsistent.

## 3.2. Channel-aware based safety messages prioritization scheme

This schemes [10], utilized received signal strength of BSM, to reduce the security issues of using mobility information in BSM, to cluster incoming messages into five fixed safety areas using the K-means clustering algorithm. Then assign BSM according to their safety areas and verify the messages depending on their arrival time.




[10]

*Figure 3.1 Multi-Level Priority Queue for Channel aware BSM verifications scheme.*

The above figure is the architecture of the existing [10] Channel aware BSM verifications scheme. We have summarized their work as follows. The following are steps for BSM to get verification time included in the existing [10] scheme.

***BSM clustering using K-means algorithm****:* in this first step, the incoming BSMs clustered into different safety areas with the help of K- Means clustering algorithm with BSM's received signal strength. There is a fact the received safety messages have different signal strengths in such a way that greater the distance between a vehicle [11]. The safety areas are mainly defined based on the requirements of the ITS safety applications that will require the classification of incoming BSMs according to the received signal strength of the transmitting vehicles. The classification of incoming BSMs according to five (k = 5) main SAs (Safety Areas) are summarized in the following table 3.1:

*Table 3-1. BSMs classification into five SAs in existing scheme*

| **Safety areas** | **Transmitter-receiver distance** |
|---|---|
| Safety Area 1 [SA1] | 0 – 50meters |
| Safety Area 2 [SA2] | 50 – 100meters |
| Safety Area 3 [SA3] | 100 – 150meters |
| Safety Area 4 [SA4] | 150 –200meter |
| Safety Area 5 [SA5] | >200meters |

**BSM dispatching into Multi-Level Priority Queue (MLPQ):** In the second step, the Multi-Level Priority Queue (MLPQ) includes a set of k Safety Area Queues (SAQ). They ranked from the highest to the lowest priority according to the considered SAs. Each $SAQ_l$ ($1 \le l \le k$) is responsible for storing the incoming BSMs which are associated with the safety area $SA_l$. Hence, BSMs generated by vehicles located within the same safety area, $SA_l$, will be grouped all together into the same $SAQ_l$, and will be processed based on their priority ($l = 1$ being the highest priority level).

**BSM scheduling for verification:** In the final step, BSMs extracted from the MLPQ to be verified by using the First Come First Served (FCFS) scheduling algorithm. The FCFS always checks the highest priority Safety Area Queue (SAQ) which is $SAQ_l$ (i.e. $l = 1$) for BSMs stored in the ready queue within their assigned safety areas. If a queue is empty, it will check the immediate lower level queue, until a BSM is found and extracted. In general, however, scheduling techniques do not give priority to messages in the buffer according to the demands of ITS application which recommend that nearby vehicle's BSM need to get verification time before far vehicles even within their corresponding safety areas.

To overcome the specified problems of the existing prioritization scheme, we proposed a novel Enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in VANETs. Our proposed Enhanced Intra Safety Area Queuing ranks the incoming safety messages depending on transmitter-receiver distance and arrival time (if distance values of two transmitters similar) to extract the messages within their corresponding safety areas to get verification time. So nearby vehicle's BSM always get verification time before other farthest. Therefore, in our proposed work, the waiting time of nearby vehicle's BSM for verification in the security queue always low, and also the cooperative awareness accuracy between neighboring vehicles also improved compared to existing schemes.

*Table 3-2. Comparison of the existing related works*

| **Paper** | **Author & Year** | **Title** | **Metrics considered** | **Gap** |
|---|---|---|---|---|
| [8] | Biswas et al, (2012) | Relevance-based verification of VANET safety messages | Acceleration/deceleration, and current speed<br><br>BSM pass through three bloom-filters (Ranks)<br><br>BSM assigned to each Ranks verified randomly | -Random verification of BSM in each rank<br>-important BSM will not be verified |
| [11] | Biswas et al (2013) | A cross-layer approach to privacy-preserving authentication in WAVE-enabled VANETs | Random | Verification probability result in important BSM will not be verified |
| [9] | J. H. Cheon et al (2007) | Fast batch verification of multiple signatures | Batch | The single invalid message will lead to dropping all |
| [12] | Sam Banani et al (2018) | Verifying safety messages using relative-time and zone priority in vehicular Ad hoc networks | Location, velocity, direction | Complete reliance on mobility information will lead to security issues |
| [13] | Sam Banani et al (2019) | Safety Message Verification Using History-Based Relative-Time Zone Priority Scheme | Location, velocity, direction, BSM-history | Complete reliance on mobility information will lead to security issues |
| [10] | E. B. Hamida et al (2017) | Channel-aware ECDSA signature verification of basic safety messages with K-means clustering in VANETs | Received Signal Strength, K-Means algorithm | Scheduling technique do not give priority to messages in buffer |

## 3.3. Summary

Generally, when we see the current safety message prioritization scheme in VANET, they have their own significance and limitation. As we express above the Channel-aware based BSM verification scheme because utilized arrival time metric to extract BSM for verification, will *increase* the waiting time of nearby vehicle's BSM in the security queue for verification & inefficient awareness accuracy between neighboring vehicles in the transmission range, even within their corresponding safety areas.

Thus, it will have an impact on the requirement for an end to end safety ITS application. In general, result in an inconsistent prioritization approach. Based on the literature, our proposed solution utilizes distance between transmitter and receiver parameter to cover the following points:

- ✓ Design a novel Enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme on highway VANETs scenario.
- ✓ Design novel architecture that is expected to minimize waiting time for nearby vehicle's BSM within their corresponding safety areas and improve cooperative awareness accuracy between neighboring vehicles.
- ✓ Develop algorithms for safety message prioritization by using the distance between transmitter and receiver to effectively extract safety messages from the Multi-level priority queue for verification.

# Chapter Four

# Design of the Proposed Solution

## 4.1. Overview

First Come First Served (FCFS) sometimes called First-In-First-Out (FIFO) is the scheduling algorithm that is a baseline for WAVE standard which gives no priority to received BSMs in a buffer. Additionally, the scheduling technique (i.e. FCFS) which has been utilized in the existing [10] scheme, used the arrival time of incoming BSM, to extract the messages within their corresponding safety areas from Multi-Level Priority Queue (MLPQ) to get verification time. However, arrival time in [10] cannot prioritize the messages according to the vehicle's status from the receiving vehicle. Hence, it could result in inefficient and inaccurate awareness between neighboring vehicles. Thus, the scheduling technique to message picking from MLPQ for verification, do not satisfy the demands of ITS safety application, which recommend that nearby vehicle's BSM need to get verification time before far vehicles even within their corresponding safety areas (SAs).

As we have proposed in Chapter one under Section 1.2, in this chapter, we are going to design the enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in VANETs. Our main objective is to design the safety message ranking scheme to improve the existing channel-aware BSM verification scheme that allows BSM to get verification time within their safety areas. In our proposed solution, we consider the transmitting-receiving vehicle distance and BSM arrival time. Note that, we are using these two parameters differently from all existing approaches. That is since the road has been classified into fixed five safety areas with help of BSM's received signal strength, the security problem with using information contained in BSM for messages clustering is now reduced. So that we are using this information to rank BSM within secured clustered safety areas, unlike, all the old BSM prioritization schemes [8, 12, 13]. Additionally, in our thesis, we have focused on three metrics such as queuing delay, the number of packets verified per distance, and awareness accuracy per distance. We will discuss in detail how to obtain and compute the distance value of transmitting and receiving vehicle and finally analyze our work with respect to the three metrics.

We have been motivated to propose our new scheme, by assuming that, if we consider the advantages of the two common existing approaches, the safety messages prioritization scheme will be efficient and reliable. To achieve this, we classify the highway road into different safety areas (SA) using BSM received signal strength (i.e. channel aware based BSM verification scheme), so less vulnerable to security threats and then, we rank BSM within their safety area using transmitter-receiver distance (i.e. vehicle status based BSM verification schemes), so the receiver is aware of transmitting vehicle's status during ranking the BSMs. As a result, the modified safety message prioritization scheme for verification will be secured and efficient than the existing one.

Therefore, in our enhanced Channel-aware based BSM verification scheme, we have improved one component on the existing [10] scheme. The component was a ranking module for extracting safety messages, so that, the messages can get verification time according to their relative proximity to the receiving vehicle. To improve queuing delay for nearby vehicle's BSM within their corresponding safety areas, we proposed a BSM ranking scheme using transmitting vehicles and receiving vehicle distance and also using BSM arrival time if two vehicles have equal distance value. In the following section, we are going to present the general architecture of our Enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in VANETs.

## 4.2. The Architecture of the Proposed Solution

In our novel proposed solution, we enhanced BSM queuing within the safety area for a Channel-aware-based BSM verification scheme to allow messages picked for the verification algorithm. It considers the safety issues during BSM extraction for verification (using ECDSA). The proposed solution's architecture shows the flow operations of the proposed work in detail. The operation includes ranking safety messages according to their assigned safety areas (SA). Our new ranking algorithm uses transmitter-receiver distance to pick the messages from a multi-level priority queue (MLPQ) for verification. Additionally, in case, if two vehicles have equal distance value, we used BSM arrival time as another ranking parameter. In our proposed solution, vehicle's BSM with the closest distance from receiving vehicles always get verification time before the farthest vehicles. Hence, the queuing delay always low, packet drop will be minimized and since more packets for the nearest vehicles are verified, the awareness accuracy improved between neighboring vehicles.

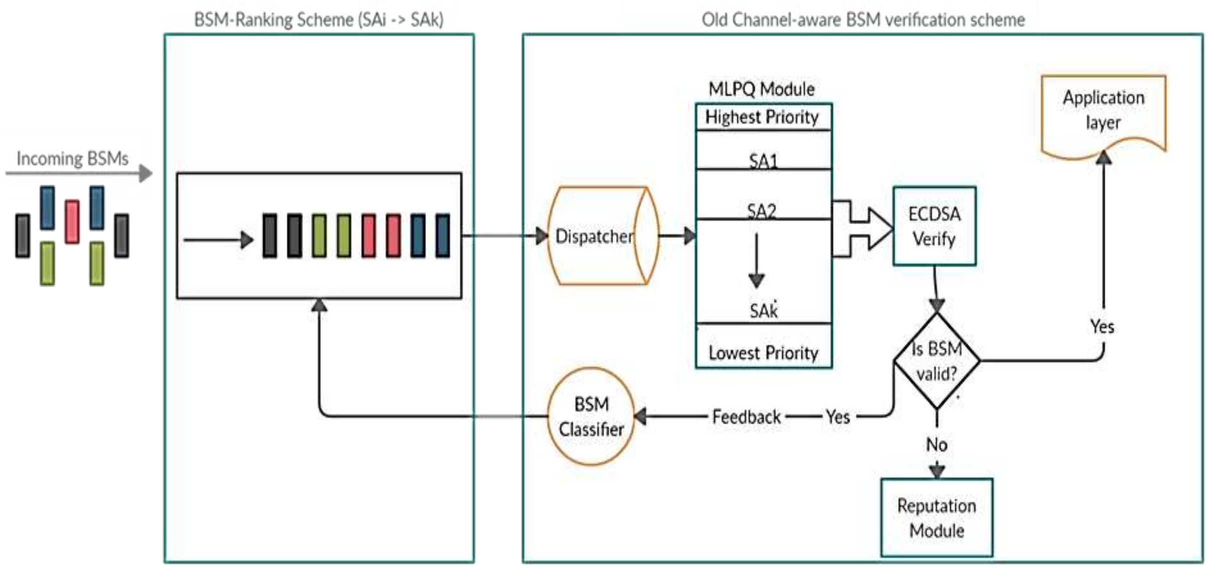



*Figure 4.1 Architecture of proposed solution (ISAQ)*

The above architecture shows how our enhanced safety messages prioritization scheme for verification in highway VANETs works. We have modified the way the message is extracted from the MLPQ module. In the existing scheme [10], it was a scheduling technique that gives no priority to received BSMs in a buffer so that, it does not satisfy the demands of ITS safety application since it doesn't prioritize the message in the buffer (in our case, 'queue'). Therefore we have replaced that with a ranking system to prioritize BSM within their corresponding safety areas to rank the BSM within their safety areas based on their relative proximity from the receiving vehicle. We have used (transmitter-receiver distance and message arrival) as parameters for ranking so that BSM can be picked from the MLPQ module for verification with respect to their importance for ITS safety application. This means it allows for the nearby vehicle's BSM to get verification time within less delay, so that awareness accuracy between vehicles will be improved. In the following subsection, we are going to elaborate on all the components explained above.

## 4.3. The Proposed BSM prioritization scheme

### 4.3.1. BSM-Classification

During the safety message ranking approach, after safety messages have been clustered into five fixed number of safety areas using K-Means Clustering algorithm with BSM's received signal strength, then BSM have been assigned according to safety areas which have highest to lowest safety area priority queue. The main advantage of clustering the BSM within the highest to lowest safety area priority queue using their received signal strength, it makes the scheme less vulnerable to message tampering or altering attack, hence there is no way BSM within the lowest priority safety area queue gets verification time before BSM within highest priority safety area queue. Simply undeserved priority will not be served.

BSMs are classified into five main clusters based on received powers to represent the considered safety areas. The BSM received power value corresponds to the safety areas are shown in Figure 4.2.

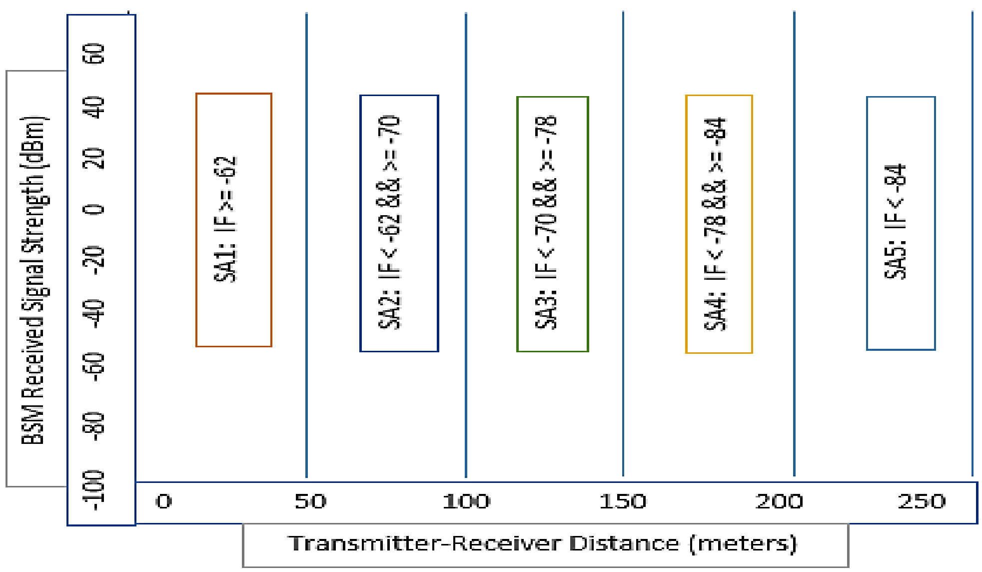



*Figure 4.2. BSMs Classification into five Safety Areas using Received Signal Strength*

### 4.3.2. BSM-Ranking

Now to rank the BSM within their assigned safety areas to extract them from the MLPQ module for verification, first we accepts the safety areas created by BSM-Classifier, then applied our ranking approach on coming BSMs corresponding to their safety areas. To achieve this, we applied transmitter-receiver distance and in addition, in case if two/more vehicles have similar distance value from the receiving vehicle, we used BSM arrival time as [10]. To clarify our work to select ranking parameters, we consider the following two scenarios.

**Scenario 1:** Assume the distance value of two or more transmitting vehicles within the same safety area (SA) varied. In such a case, we do rank based on distance values to give verification time for the most nearby vehicles than farthest ones. As we observe from the following figure 4.2, there are many vehicles in the transmission range with different distance values from the receiving vehicle.

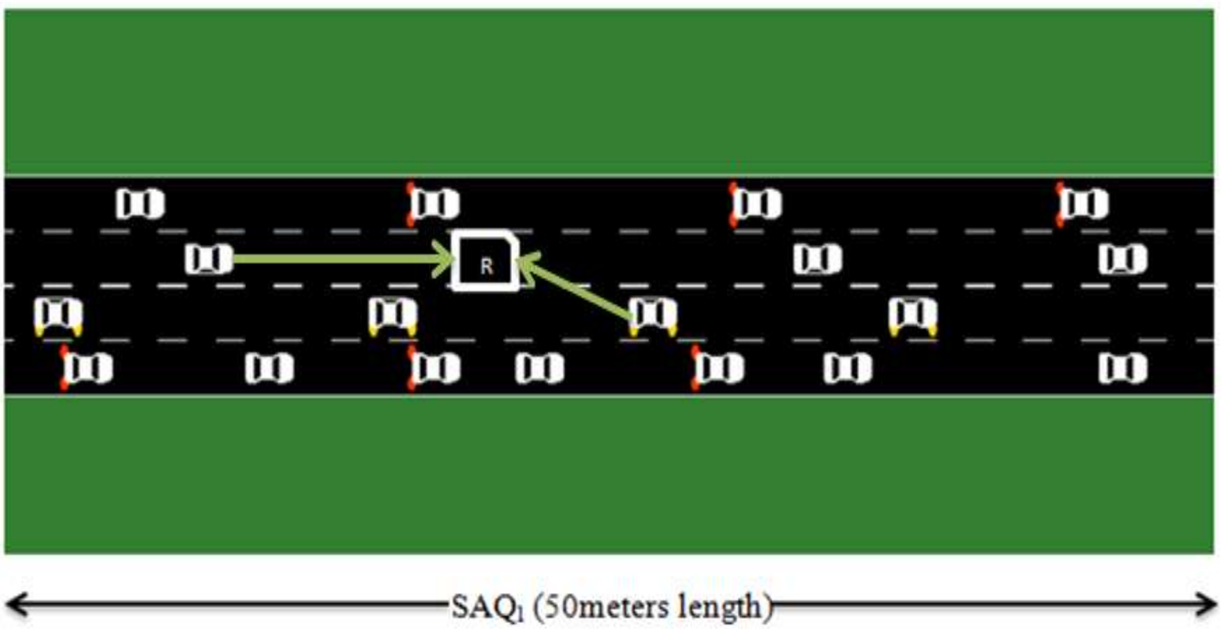



*Figure 4.3 ISAQ's Scenario when distance value varied*

To achieve this, we have computed two vehicles' distances with the help of the (x, y) position of the transmitting and receiving vehicles using the following equation (1), which is a two-dimensional distance formula [10].

$$Dist = \sqrt{(x2-x1)^2 + (y2-y1)^2} \quad \text{.......................................equation (1)}$$

Where 'Dist' is the distance, (x1, y1) represents the transmitting vehicle's position, and (x2, y2) represents the position of the receiving vehicle. After we compute the distance between the two vehicles, our ranking takes the distance value as an input which is in the range of 50meters. Then small distance value gets the highest priority in the queue and the far distance values will get the lowest priority.

**Scenario 2:** Assume the sometimes, some transmitting vehicles can have similar distance value even within their same safety area (SA) varied. In such a case, we applied ranking based on BSM's arrival time values to give verification time. If we look at the below figure 4.3, there are some vehicles with the same distance values from the receiving vehicle.

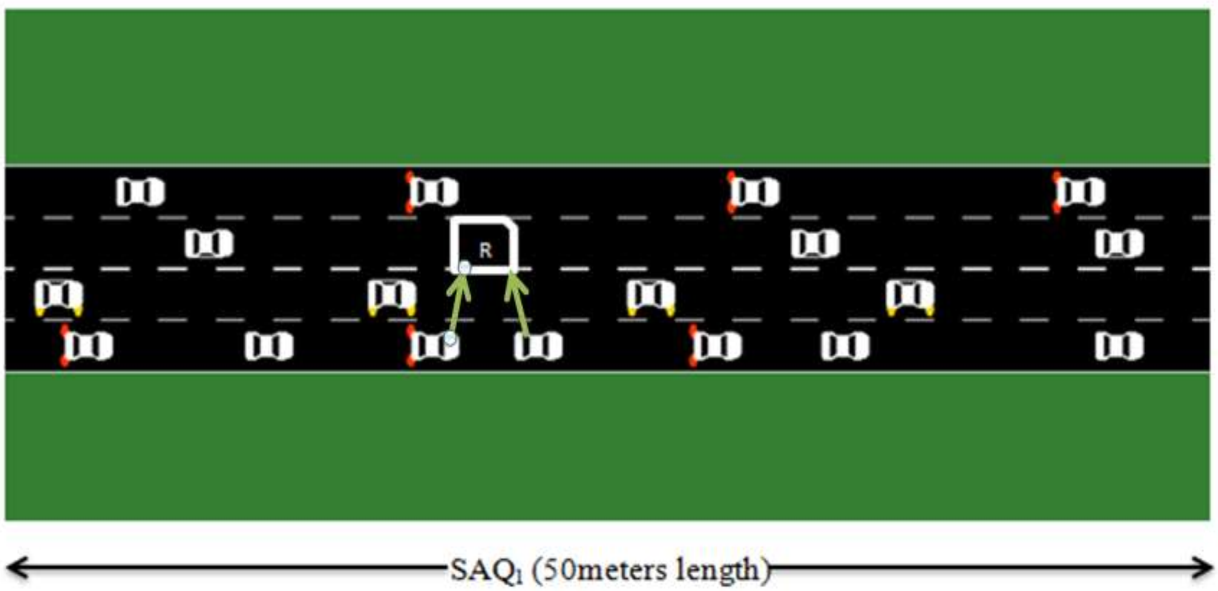



*Figure 4.4 ISAQ's Scenario when distance value similar, use the arrival time*

As the above figure shows that if the distance value of the transmitting vehicles similar, at this time we used the arrival time of messages in the ready queue to pick the BSM for verification. To achieve this, we have calculated the arrival time of each BSM using equation (2) below:

*Arri_T = Tag. Now ()..............................................................equation (2)*

Where '*Arri_T*' represents BSM's Arrival Time. Besides, '*Now ()'* shows the time at which the safety message reached in the receiving vehicle's buffer. And '*Tag'* refers to the broadcasted BSM. Our ranking scheme will always give nearby vehicles (closest to the receiver) verification time before the far away from vehicles even if they are found within their safety areas. So that, when the distance of all vehicles within their safety area ranked from closest to farthest vehicles, BSM waiting time in the queue for verification will be in accordance with their proximity from the receiver. But if there is the possibility for two/more vehicles with similar distance values, we used BSM's arrival time and then, the awareness accuracy between the neighboring vehicles improved.

In general, depending on these two parameters (distance and arrival time), we allow BSM within their corresponding safety areas to get verification time. The following Figure 4.3 shows the flow of operation for our proposed work.

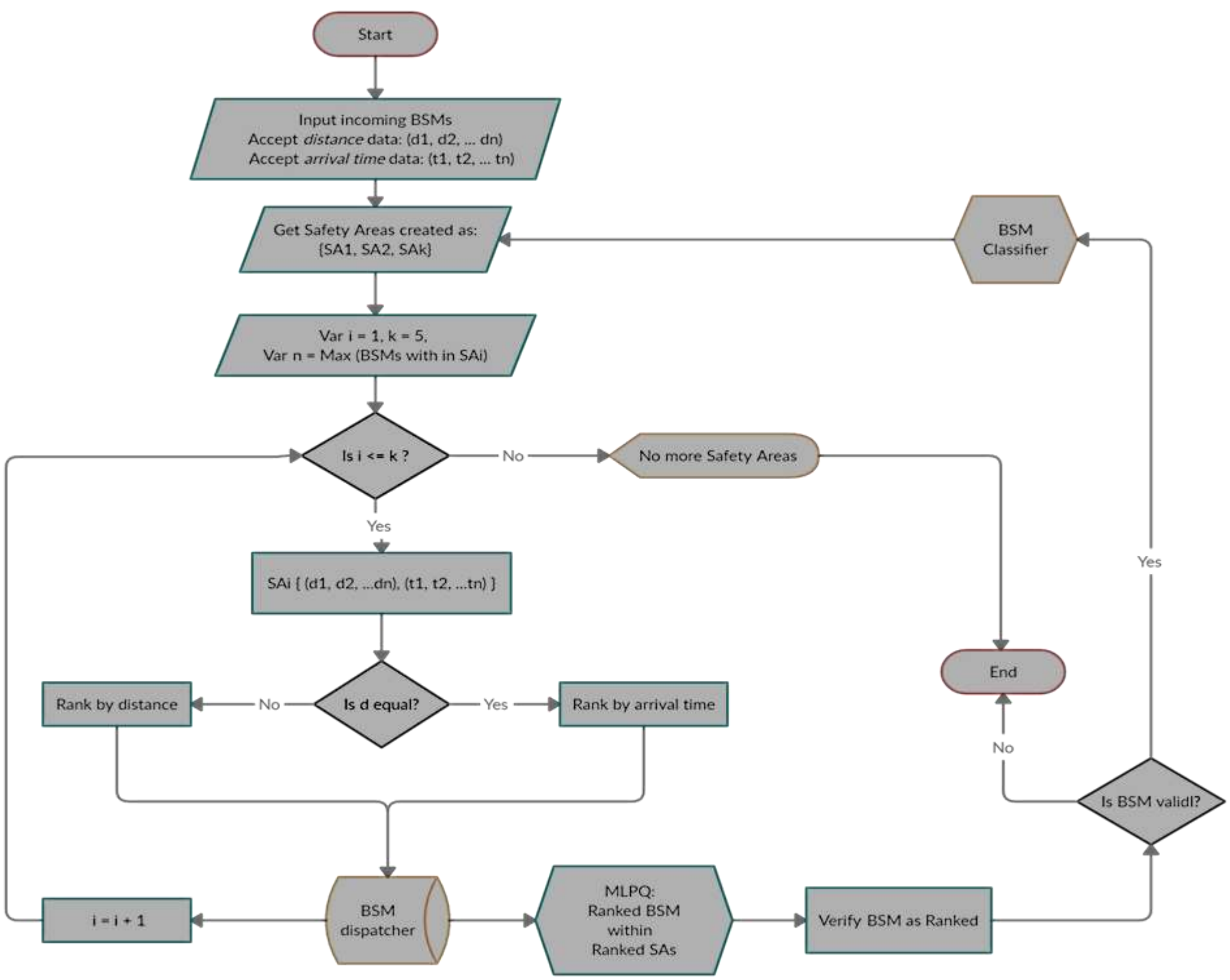



*Figure 4.5 Flow chart for BSM ranking within their SA*

The above Figure 4.5 shows that how our proposed solution operates. At the first step, it takes safety areas created (five priority classes) from BSM Classifier. Then assign incoming BSM into their corresponding safety areas based on their received signal strength. Within each safety area, it computes takes the distance between transmitting and receiving the vehicle from their (x, y) positions values using of two-dimensional distance formula which has been discussed in equation (1). Then the ranked BSM will be dispatched into the Multi-Level Priority Queue Module. The module contains Safety Area Queue (SAQ) that has the responsibility to store all BSM corresponds to the safety areas they have assigned based on their signal strength. The SAQ is also arranged from highest priority (SAQ1) to lowest priority (SAQk). So BSM within SAQi to SAQk being ready for verification. Finally, every BSM gets verification time based on their rank within their corresponding Safety Area Queue.

If we look at the following table 4-1, the effect of extracting BSM from MLPQ in our proposed work by considering the two scenarios above and also their waiting delay in the queue corresponds to their rank have shown. Let assume (BSM1, BSM2, BSM3, BSM4, and BSM5) be the number of safety messages assigned to the first safety area (SA1) whose distance between 0 – 50 meters. Note that processing (verification) time for single BSM using ECDSA algorithm on average equal to 5ms [13].

*Table 4-1. The effect of our ranking scheme according to two scenarios (1 and 2)*

| **Incoming Messages** | **D & AT value** | | **Ranking order and BSM's waiting time** | |
|---|---|---|---|---|
| | **D(m)** | **AT (ms)** | **Rank** | **Waiting Time** |
| BSM1 | 0.36 | 1930022 | 1st | 0 |
| BSM2 | 1.79 | 1078020 | 3rd (by AT) | 10 |
| BSM3 | 1.79 | 639020 | 2nd (by AT) | 5 |
| BSM4 | 3.478 | 4060416 | 4th | 15 |
| BSM5 | 14.41 | 4486508 | 5th | 20 |

Table 4.1 shows how our ranking approach allows all five messages get verified that. It accepts both the distance and arrival time values. If the distance between transmitter and receiver is different, we do rank by using distance values. But, in case if the distance values of the transmitter are the same, we do rank by using the message's arrival time similar to [10]. Hence in our new scheme, both the waiting time and ranking accuracy of every BSM is with respect to the

rank (i.e. the smaller the distance, the lower the waiting time & smaller rank and the reverse is true.

**Algorithm**: *BSM ranking within their safety areas (SAs) by proposed solution*

**Input** ===> incoming BSMs // accept incoming BSMs
**Input** ===> (d1, d2, … dn) //set of distance values computed
**Input** ===> (t1, t2, … tn) // set of arrival time all BSM
**Get** ===> SA from BSM classifier

1. ***for*** i= 1 to of k ***do***
2. SAi = *get* (d1, d2, …dn) // distance values of BSM within SAi
3. ***do***
4. ***if*** *distance no equal*
5. *rank_asc(d1, d2 , …dn)* // rank by distance
6. ***end if***
7. ***else if*** *distance equal*
8. *rank_asc(t1, t2,…tn)* // rank BSM arrival time
9. ***end else if***
10. ***while*** *(not empty)*
11. ***return i++*** **//** check until the last (lowest priority) SA

According to the above algorithm, in the first step, we accept the BSM clustered into fixed five different safety areas from the BSM classifier (K-Means clustering algorithm). Then, starting from SA with highest to lowest priority, we receive transmitter-receiver distance and also BSM arrival time values. Finally, we do rank the BSMs according to the two scenarios (1 and 2) we have considered until no queue is empty (no BSM assigned to each safety area).

## 4.4. Summary

In this Chapter, we have presented the overall architecture of our BSM prioritization scheme for verification in VANETs. So, when we see the existing work, they have their own advantages and limitation. As we explained in the above sections, the existing BSM prioritization schemes does not fully fill the requirement of ITS, especially for Cooperative deriving applications that need to be less vulnerable to a variety of security threats, and also best cooperative awareness accuracy between neighboring vehicles need to achieve. Due to these requirements, it's important to design new safety messages prioritization scheme for verification in highway VANETs scenario.

# Chapter Five

# Implementation and Evaluation

## 5.1. Overview

In the previous Chapters, we have shown that designing Intra Safety Area Queue for channel-aware BSM verification scheme in VANET can solve the problem of messages waiting time in queue for verification and also improve awareness accuracy between neighboring vehicles in accordance with ITS safety applications requirements. Therefore, to achieve this objective we have presented in Chapter 4, the designs of Intra Safety Area Queue for channel-aware BSM verification scheme to rank the BSM using transmitter-receiver distance and BSM arrival time, which can always allow closest vehicle's message, get verification time within less delay. In our proposed solution, we have considered the highway scenario of highly dense VANET environments.

Due to the excessive costs of VANET entities (e.g. vehicles) and the wireless access network technologies in real-world testbeds, our proposed ranking scheme has been implemented and evaluated using a simulator. So, we have used the NS3 simulator to trace safety messages received by each vehicle in the network, within their transmission range. And for message ranking, we used Jupyter notebook (i.e. python platform). A detailed description of the implementation of our proposed work is presented under sub-sections of this Chapter. Section 5.2 describes the development environment employed to implement the scheme. In section 5.3, the simulation experiment and evaluation result are described. Finally, section 5.4 presents the summary of the Chapter.

## 5.2. Development and Simulation Tools

The selection of development environment and simulation tools that were used for implementation and evaluation of our proposed solution is described in this Section. We have used different simulation tools by integrating them to implement our proposed solution. VANETs simulators and Python platform for data analysis (in our case, for message ranking). VANETs simulation requires two types of simulation components those components are Network and mobility. First, different types of vehicle traffic mobility simulators have been discussed in Section 5.2.1 and then basic types of VANETs network and integrated simulators as shown in Section 5.2.2. Furthermore, all traffic mobility and network simulators have varied factors to be considered in simulating a VANETs environment. Finally, for data analysis (for BSM classification and ranking) different tools have been presented in section 5.2.3.

### 5.2.1. Traffic Mobility Generators

The realistic vehicular mobility traces to be used in network simulator as input is required in these types of simulators generates. The comparative studies on VANETs traffic mobility simulators presented by different scholars. In [50], they described and analyzed VANETs traffic mobility simulators like SUMO, VanetMobiSim, MOVE, FreeSim, and City mob. Their analyses are based on features like freeware, portability, XML based trace support, GUI support, ease of use, user-defined map, and available examples. After comparison of the simulators in [50], SUMO and VanetMobiSim are recommended as the best choices when supporting all traffic models, and good software features are considered for research work. Based on this assessment, SUMO is highly portable, functional across various scenarios, designed for use in traffic strategies and enhancement of route layout.

SUMO [54] stands for Simulation of Urban Mobility (SUMO), it is an open-source, highly portable microscopic road traffic simulation package that deals with a very large number of nodes in VANET. It can be used on most operating systems. Because of high portability and its GNU General public license, SUMO has become more popular and most widely used in vehicular ad hoc networks. It has progressed into a full-featured suite of traffic modeling utilities that uses its own formats for traffic demand generation and road networks and routing utilities. The main advantages of SUMO are that it is OpenGL GUI based; generates real traffic mobility, highly portable, open-source, easy simulation set-up, portable libraries, collision-free movement, imports different formats, and a large number of the map defined for better understanding. Therefore, we have selected SUMO as a traffic mobility generator in our proposed work.

### 5.2.2. Network Simulators

The comparative studies on many network simulators are presented by different scholars. In [51] the scholars described and examined network simulators like OPNET, NS-2, GloMoSim, and QualNet. The analyses for network simulators are done based on their features like GUI support, distributed simulation support, scalability, antenna support, and multiple wireless technologies support. Based on this evaluation, OPNET and QualNet have supported all the above-mentioned features though they are not free and do not support the real mobility pattern of vehicles. However, NS-2 does not support multiple wireless technologies. The evaluations done for network simulators are depending on their features like language support, weaknesses, and strengths. The results of the examinations are almost similar to the general assessment outcomes mentioned in [52].

On the other hand, NS-3 [53] is a discrete-event network simulator, directed primarily for educational and research use. It is free software, licensed under the GNU GPLv2 license, and is publicly available for research, development, and use. The NS-3 project has started in 2006, it is not a backward-compatible extension of NS-2; it is a new simulator. The two simulators are both written in C++ but NS-3 is a different simulator that does not support the NS-2 APIs and it allows coding in C++ and Python to simulate a simple and complex networking scenario. NS-2 some models have already been exported to NS-3, and the NS-3 project will continue to maintain NS-2 while NS-3 is being built, and will study transition and integration approaches.

So, a survey in [53] showed that NS-3 (Network Simulator version 3) has the capability to handle large-scale scenarios, with even 10,000 nodes, and support multiple wireless interfaces in a single node. Furthermore, it is open-source with GNU licensed. Based on our have observed from the comparative studies and analyses presented in [51], [52], [53], SUMO which stands for Simulation of Urban Mobility, is the best choice as a traffic mobility generator that provides a realistic mobility model, functionality in different scenarios and high portability of trace file for VANETs. While from VANETs network simulators, NS-3 is the preferred one regards to supporting multiple wireless interfaces in a single node and freely available or non-commercial.

Additionally, NS-3 can support two kinds of visualizers: PyViz and NetAnim. PyViz is standing for Python Visualization, a default live simulation tool of NS3 that programmed in Python script but it is not attractive. While NetAnim is shorted for Network Animator, an offline animator based on the Qt toolkit and uses an XML trace file generated by NS-3. Hence, we have used the NetAnim visualizer to emphasize the user interface of the simulation. In general, we have selected NS-3.30 as a network simulator for implementing vehicle communication and storing each vehicle's received packets as an input to our ranking scheme.

#### 5.2.3. Data Processing Tools

The comparative studies on data processing and analysis platforms such as MATLAB and Python have been presented in [55]. The studies show that MATLAB is widely known as a high-quality environment for any work that involves arrays, matrices, or linear algebra. Python is newer to this arena but is becoming increasingly popular for similar tasks. Python is a mature language developed by hundreds of collaborators around the world. Both MATLAB**,** Python is an interpreted language. This means that their code can be ported between all of the major operating system platforms and CPU architectures out there, with only small changes required for different platforms.

According to a survey by [56], an important philosophical difference in the MATLAB and Python comparison is that MATLAB is proprietary**,** closed-source software. So, a license to use MATLAB is quite expensive. On the other hand, Python is free and open-source software. So you can also download, look at, and modify the source code of Python. This is a big advantage for Python because anyone can pick up the development of the language. A very popular Python distribution, particularly for math, science, engineering, and data science applications, is the Anaconda distribution. The main reasons for the popularity of Anaconda are i) Anaconda distributes pre-built packages for Windows, macOS, and Linux, which means that the installation process is really easy and the same for all three major platforms. ii) Anaconda includes all of the most popular packages for engineering and data science type workloads in one single installer.

Therefore, in our proposed solution, we have used python and its platform like Jupyter Notebook to clustering, ranking and analyzing of received safety messages by a particular receiving vehicle.

## 5.3. Prototype Implementation

To generate mobility traces for vehicles by SUMO traffic simulator to model a highway scenario designed on NetEdit SUMO built-in network editor. A road network of 1km×1km is used. The vehicle density is set to 200vehicles/km to create a dense network. For a generation of mobility models, conventional vehicles/cars have been used.

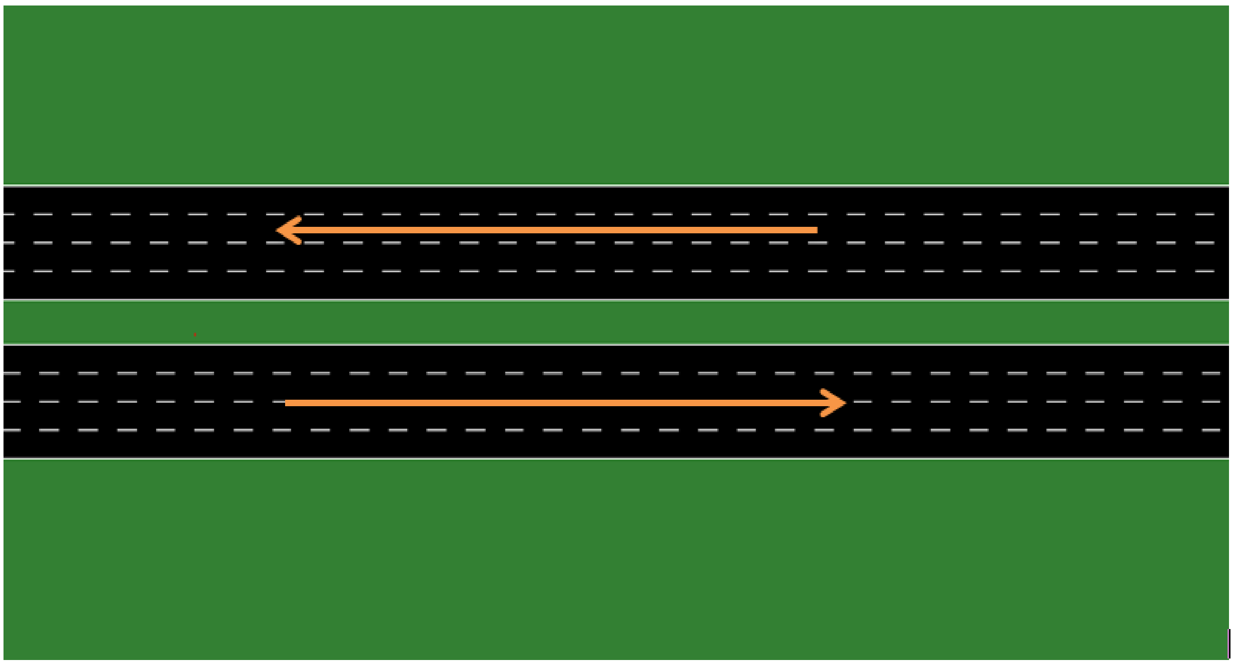


*Figure 5.1 Design of the highway scenario we consider*

Then, after completion of our design, we simulate on sumo-GUI to check the traffic flow on our road scenario. We have summarized the parameters of mobility generation in Table 5.1 below.

*Table 5-1. Mobility Model Generation Parameters*

| Parameter | Value |
|---|---|
| Type of street | Highway |
| Road length | 1km |
| Nodes (vehicles) number | 300 |
| Number of lanes | 4 |
| Simulation time | 200 seconds |

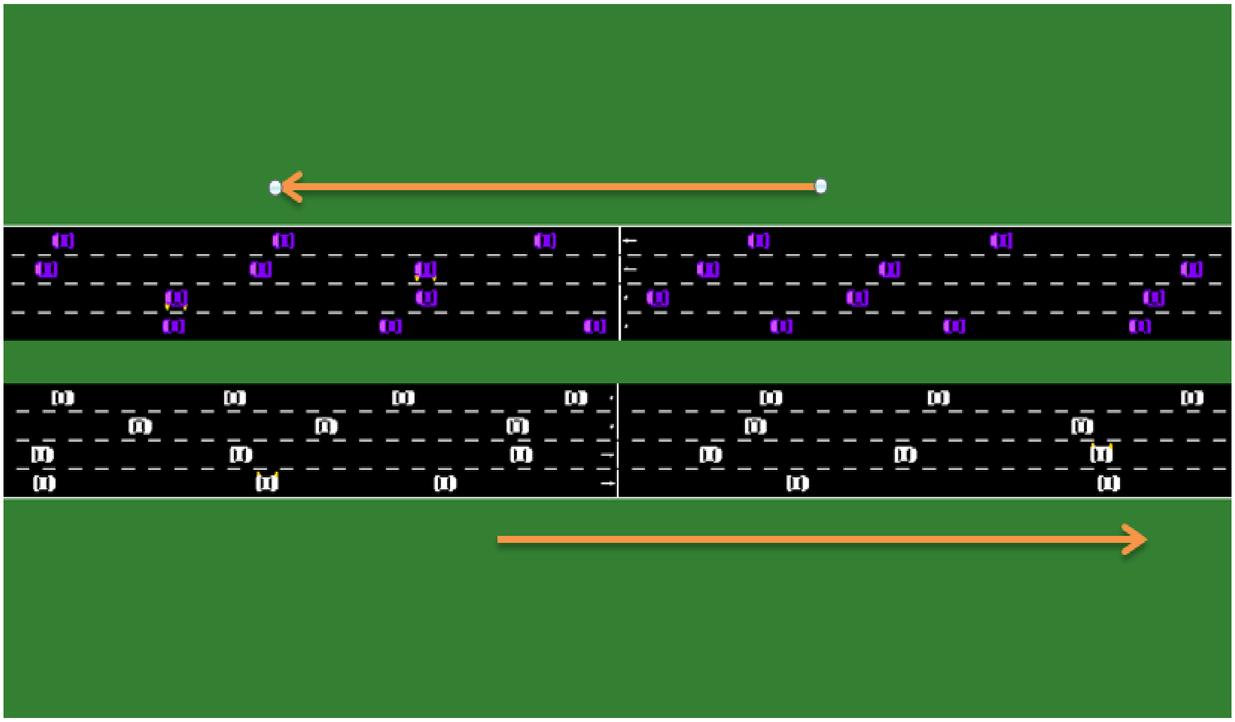


*Figure 5.2 The sample of traffic mobility model in our highway scenario*

Trace files (XML files) generated by SUMO can be exported to different network simulators such as NS-3. However, NS-3 is programmed with C++ and Python, so it primarily used TCL and py extension files. Therefore, before the actual network configuration of vehicles, we have converted the generated trace file of vehicles mobility model to (.tcl) file which supported by NS-3 network simulator as shown in Fig 5.3.

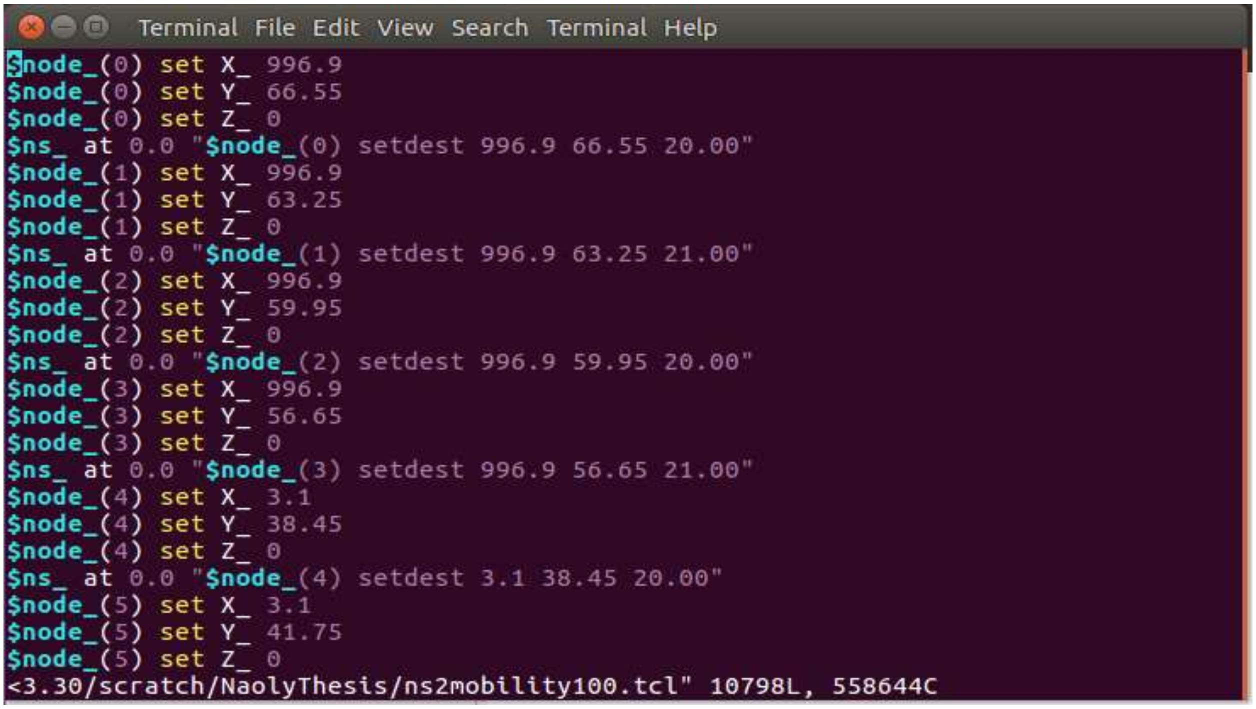



*Figure 5.3 The sample Generated Mobility Model of vehicles in Tcl File*

After we completed the generation and conversion of vehicles' realistic mobility model, we have preceded to the next step which is the configuration of the vehicles. In this step, directly we imported the mobility Tcl file looks like on the above Figure 5.3, to use the generated vehicles mobility in the NS-3 simulator. Then we proceed to the configuration of WAVE Interface and BSM application on Vehicles. This step is the simulation of vehicle communication on NS-3. Each vehicle has a configured WAVE setup. We used WaveHelper, QosWaveHelper [57] of NS-3 helpers are implemented on PHY and MAC layers of vehicles respectively. BSM applications are installed on devices like BSM format and information in the BSM. We have created our NS-3class to extend the built-in application class and program the way nodes broadcast and receive BSM accordingly.

*Table 5-2. Attributes of WAVE Interface on Vehicles*

| **Variable** | **Value** |
|---|---|
| Transmission range (power) | 33dBm (for non-governmental) |
| Channel width | 10 MHz |
| Data rate | 6 Mb/s |
| BSM interval | 100 ms |
| BSM size | 200 bytes |

The general using NS-3 simulation in our proposed work is to get a vehicle packet trace. So we have stored the packet (BSM) they received from their neighbor in the transmission range in (.CSV) file. The sample of safety message received by specific vehicles has been shown in Figure 5.4 below:

| A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|
| Brdcast_Time(ms) | Recv_Id | Send_Id | Send_X.Pos | Send_Y.Pos | Arrival_Time | Received_Power | Recv_X.Pos | Recv.X_Pos | TR_distance |
| 49000 | 3 | 1 | 28.0963 | 63.25 | +49000966556.0ns | -47.8939 | 14.6775 | 59.95 | 13.81861764 |
| 49000 | 3 | 0 | 14.6485 | 66.55 | +49001831600.0ns | -38.264 | 14.6775 | 59.95 | 6.600063712 |
| 49000 | 3 | 2 | 29.3187 | 59.95 | +49002257679.0ns | -48.6478 | 14.6775 | 59.95 | 14.6412 |
| 49000 | 3 | 105 | 41.0735 | 45.0506 | +49002683938.0ns | -58.1571 | 14.6775 | 59.95 | 30.31073962 |
| 49000 | 3 | 11 | 79.0672 | 56.65 | +49003549226.0ns | -67.9597 | 14.6775 | 59.95 | 64.47420776 |
| 49100 | 3 | 110 | 3.1 | 45.05 | +49102550902.0ns | -51.8373 | 14.6775 | 59.95 | 18.86924763 |
| 49100 | 3 | 9 | 74.3115 | 63.25 | +49103402948.0ns | -67.0074 | 14.6775 | 59.95 | 59.72523718 |
| 49200 | 3 | 2 | 25.7007 | 59.95 | +49200966564.0ns | -45.2483 | 14.6775 | 59.95 | 11.0232 |
| 49200 | 3 | 1 | 24.1983 | 63.25 | +49201392573.0ns | -44.0808 | 14.6775 | 59.95 | 10.0764891 |
| 49400 | 3 | 2 | 22.0827 | 59.95 | +49400966576.0ns | -40.21 | 14.6775 | 59.95 | 7.4052 |
| 49400 | 3 | 100 | 73.6077 | 45.05 | +49403096844.0ns | -67.2579 | 14.6775 | 59.95 | 60.78468945 |
| 49400 | 3 | 8 | 65.2556 | 66.55 | +49403522888.0ns | -64.9577 | 14.6775 | 59.95 | 51.00690345 |
| 49500 | 3 | 2 | 20.2737 | 59.95 | +49500966581.0ns | -36.7017 | 14.6775 | 59.95 | 5.5962 |
| 49500 | 3 | 1 | 18.3513 | 63.25 | +49501831623.0ns | -34.9594 | 14.6775 | 59.95 | 4.938299954 |
| 49500 | 3 | 17 | 108.612 | 63.25 | +49503536383.0ns | -72.8993 | 14.6775 | 59.95 | 93.99244805 |
| 49600 | 3 | 110 | 3.1 | 45.05 | +49600966683.0ns | -51.8373 | 14.6775 | 59.95 | 18.86924763 |
| 49600 | 3 | 109 | 3.1 | 41.75 | +49601392703.0ns | -53.6071 | 14.6775 | 59.95 | 21.5703154 |
| 49600 | 3 | 1 | 16.4023 | 63.25 | +49602244767.0ns | -31.2004 | 14.6775 | 59.95 | 3.72356483 |
| 49600 | 3 | 98 | 68.1909 | 38.45 | +49603097167.0ns | -66.5728 | 14.6775 | 59.95 | 57.67091103 |
| 49600 | 3 | 100 | 77.5857 | 45.05 | +49603536228.0ns | -68.0588 | 14.6775 | 59.95 | 64.64867846 |
| 49800 | 3 | 1 | 12.5043 | 63.25 | +49801844680.0ns | -31.151 | 14.6775 | 59.95 | 3.95130336 |
| 49800 | 3 | 0 | 14.64 | 66.55 | +49802270703.0ns | -38.2719 | 14.6775 | 59.95 | 6.600106533 |
| 49800 | 3 | 18 | 107.001 | 59.95 | +49803123399.0ns | -72.6669 | 14.6775 | 59.95 | 92.3235 |
| 49900 | 3 | 109 | 3.1 | 41.75 | +49900944082.0ns | -53.6071 | 14.6775 | 59.95 | 21.5703154 |

*Figure 5.4 Sample of safety message received by node-3 with [ .CSV] format*

Then, we have imported the statistics of the received BSM from every vehicle, for Jupyter Notebook which is one of the best favorite Python platforms for data analysis. In this step, the K-means clustering algorithm has been trained on our data (i.e. distance and received signal strength), to cluster BSM into different safety areas. Then, the final classification of BSM has been done only using received signal strength into five fixed safety areas from (SA1- SA5) similar to [10]. Finally, we do our ranking scheme, within the classified safety areas (SA1 to SA5). As we have expressed in detail in previous chapters, our proposed solution prioritizes incoming BSM by using two ranking parameters (i.e. transmitter-receiver distance and BSM arrival time).

## 5.4. Simulation Experiment and Result Analysis

To analyze the performance of our proposed Enhanced Intra Safety Area Queuing (ISAQ) for Channel-aware based BSM verification scheme in VANETs. we develop a simulation model in NS-3. To do that, first, we made a simulation set up to conduct the trace of vehicle’s safety messages on NS-3. Then, we import the received safety messages for Jupyter NoteBook in order to implement our ranking scheme. Finally, we analyze and compare our scheme with the existing channel-aware-based BSM prioritization scheme.

### 5.4.1. Simulation Setup

To generate mobility traces for vehicles, we use the SUMO traffic simulator to model a highway scenario. A road network of 1km×1km is used. The vehicle density is set to 200vehicles/km2 to create a dense network. The maximum vehicle speed is taken as 22m/s. The WAVE model in NS-3 is used for BSM transmission exchange between vehicles. Each vehicle generates 10 BSMs per second with a transmission range of 300m and 6Mbps of data rate.

*Table 5-3. Simulation parameter*

| **Parameter** | **Value** |
|---|---|
| Road area | 1,000m X 1,000m |
| Simulation time | 200s |
| Transmission range of vehicle | 300m |
| Number of lanes | 4 per Direction |
| Vehicle Speed | 22 m/s |
| Packet Size | 200 bytes |
| BSM Broadcast Interval | 100ms |
| Data rate | 6Mbps |

### 5.4.2. Performance Evaluation Metrics and Results

To evaluate and compare our proposed enhanced BSM Safety Area Queuing (SAQ) for verification scheme with MLPQ-CH for verification scheme, we use different metrics. The following three metrics are used to study the performance of our scheme with other schemes:

- ✓ **Cooperative Awareness accuracy vs. each safety area**: The awareness accuracy is computed from the intersection of the actual number of N neighboring vehicles and vehicles discovered relative to their distance from receiving vehicles within each safety area. Awareness accuracy among neighboring vehicles related to the rank the message has got to be verified.

$$Coop\_Aware\_acc^{T}(i) = V_k^{T}(i)\ intersection\_of\ [N_k^{T}(i)]\ \ / VD_k^{T}(i)] \ldots\ldots\ldots\ldots eq\ (3)$$

where $V_k^T(i)$ represents the actual number of neighbors of vehicle i and $N_k^T(i)$ represents the advertised number of neighbors received by vehicle i and $VD_k^T(i)$ represents the relative vehicle distance in BSM within an area k at a certain time T.

Since the existing scheduling technique does not consider the distance between vehicles in each safety area, the cooperative awareness accuracy between vehicles is lower than our approach. Therefore, the improved vehicle awareness accuracy can improve the Quality of

Service (QoS) for cooperative awareness applications, as the vehicles within the closer distance have a higher safety concern.

- ✓ **BSM waiting time vs. distance between vehicles:** BSM waiting time is the time every BSM should wait in the security queue for signature verification. Note that, verification time for a single BSM is 5ms. Since our proposed solution extract BSMs from MLPQ based on the transmitting vehicles' relative distance to the receiving vehicle, the verification waiting time for BSM within each safety area significantly reduced, when compared to the scheduling technique in the existing scheme.

Depending on the evaluation, the performance between CH-scheduling based and our new scheme are evaluated using our simulation tool. Therefore, as we conducted from the simulation, the relation between the CH-scheduling based and our new scheme in terms of the cooperative awareness Accuracy among neighboring vehicles within their corresponding safety areas can be expressed in below Figure 5.5 as follows.

**Cooperative awareness Accuracy achieved by the two schemes:**

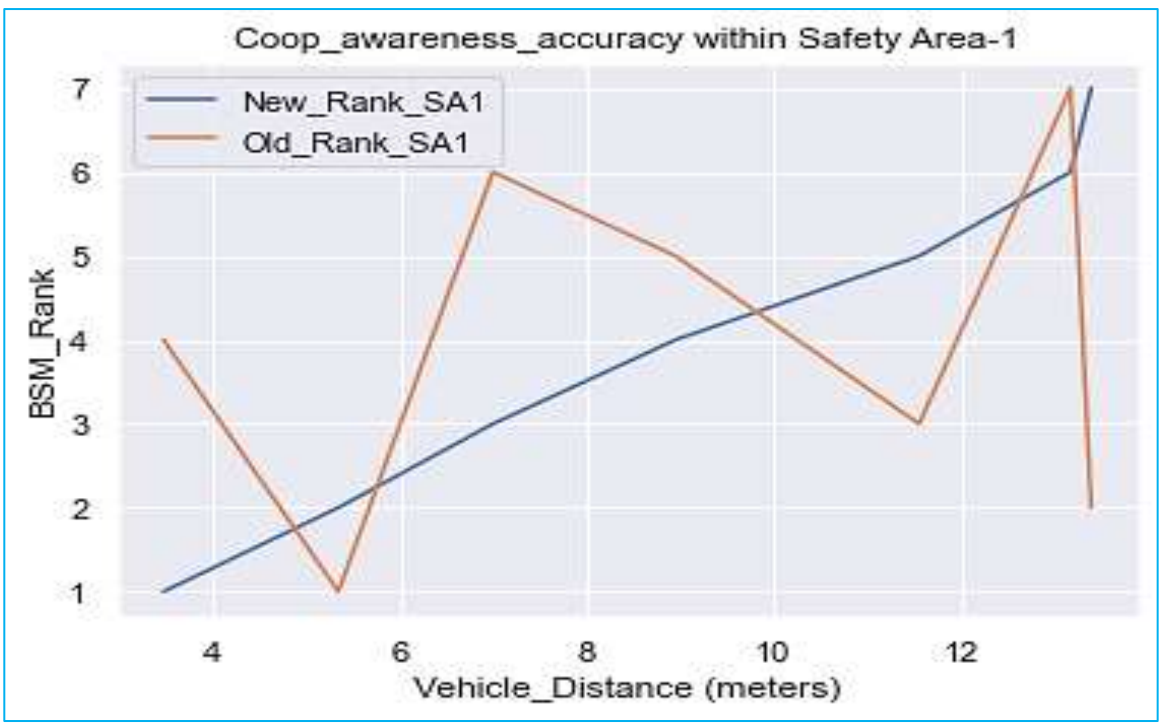



*Figure 5.5 Cooperative Awareness Accuracy among vehicles in SA1*

From the above figure 5.5, we can observe that since the BSM's arrival time do not relate to the status (closeness) of the transmitting vehicles from the receiving vehicle, sometimes the rank they will get cannot be matched according to the demands of ITS safety application in cooperative awareness applications. However, the awareness accuracy achieved by our new schemes is based on the proximity of the vehicle (closeness) from the receiving vehicle have been improved much than the existing one.

**BSM's waiting time in the queue to get verification time within their corresponding safety areas:**

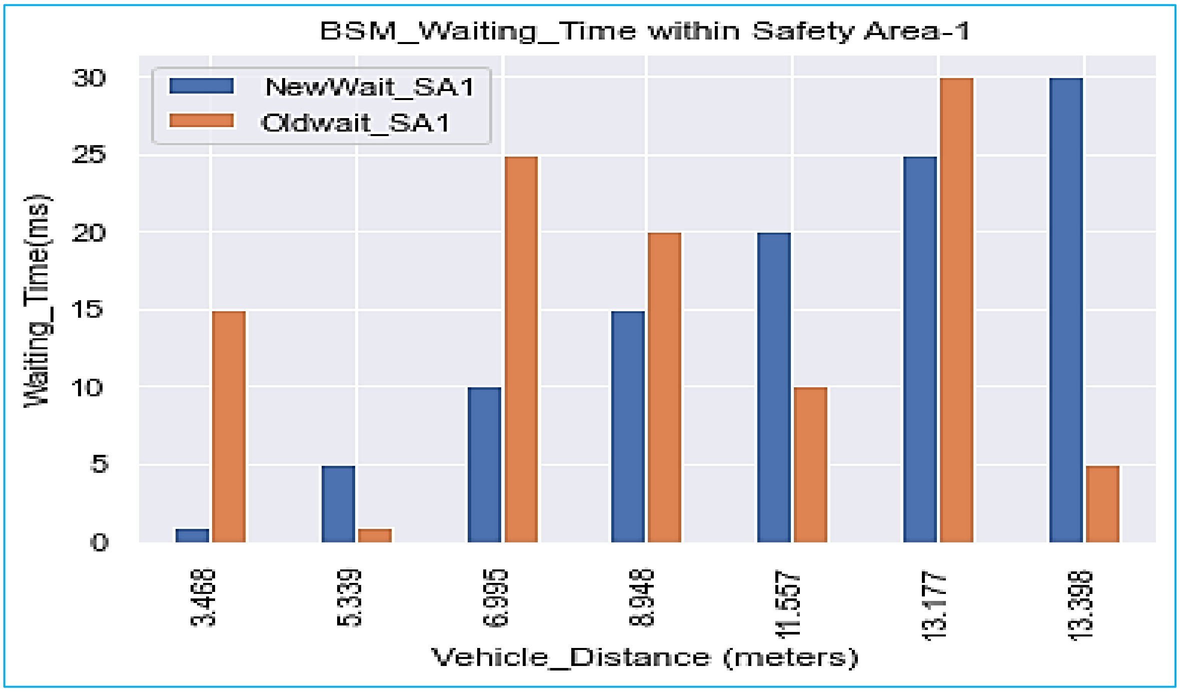



*Figure 5.6 BSM's waiting time achieved in SA1*

As the above figure 5.6 showed that the messages waiting time in the security queue to get verification time by our approach is in accordance of transmitting vehicles distance from receiving vehicle, however, since the FCFS scheduling cannot considering the status of the transmitting vehicles, messages waiting is not consistent, that means, BSM of nearby vehicles sometime low, sometime high. So, our proposed solution over perform the existing scheme.

## 5.5. Summary

In general, we did test our proposed solution by using SUMO, NS-3, and a Python platform called Jupyter Notebook. After the extensive experiments, we analyzed the performance of our propped solution, and finally, we compared our work with the existing BSM prioritization scheme. From the simulation result, we can conclude that our proposed solution achieved better in terms of BSM waiting time in the vehicles' security queue and accuracy of cooperative awareness between neighboring vehicles.

# Chapter Six

# Conclusion, Contribution and Future works

## 6.1. Conclusion

Vehicular Ad hoc Network (VANET) facilitates cooperative awareness applications by periodically sharing basic safety messages (BSMs) with the neighborhood vehicles. Since these applications impact human safety, the authenticity of the BSMs is a key requirement. However, a challenging task is that lots of BSMs are queued up simultaneously for verification, especially in a high traffic density. As a result, many important BSMs from nearby vehicles may experience significantly long signature verification delays.

Currently, there are two common approaches to address the problem of mismatch between BSM arriving rate and verification rate. Vehicle status based and channel-aware based BSM prioritization schemes for verification in VANET. The former completely rely on the BSMs mobility information to classify the road into different safety areas. However, since the mobility information in BSM can be exposed before verification, this can result in the schemes be vulnerable to different security threats. Hence, undeserved priority can be served especially during zone creation. Thus, it can impact the safety of the end-to-end ITS application. Later utilizes the received signal strength of BSM to cluster the messages into five fixed different road safety areas (SA). Then the BSM assigned to the highest safety area (SA1) will get verified before BSM within the lowest safety area (SA5). The approach reduced the security problem in former schemes. However, its key limitation is the way the BSMs within their safety areas extracted from multi-level priority queue for verification, in which, since there is no relationship between BSM arrival time and vehicle's status, the scheme does not allow the receiving vehicle to consider the status of its neighboring vehicles.

In this paper, we propose enhanced BSM queuing within the safety area for a Channel-aware-based BSM verification scheme by considering the hybrid of vehicle status and channel aware a priority-based BSM queuing scheme to reduce security issues plus the verification time of BSMs according to ITS safety application requirements. In our proposed solution, we classified the highway road into five fixed safety areas using the received signal strength of BSM (i.e. to improve security problems) and rank BSM within their safety areas using transmitter-receiver distance and message arrival time (i.e. improve safety and awareness accuracy between neighboring vehicles).

Finally, we have tested, evaluated, and proved our proposed scheme with the existing ones. The proposed scheme outperforms all mentioned evaluation metrics on the highway with high traffic density. It provides better performance in case of BSM waiting time in the security queue for verification and improved the awareness accuracy between neighboring vehicles. Thus it can be a good candidate for safety messages prioritization in VANET cooperative awareness application.

## 6.2. Contribution

The main contribution of this thesis is developing a better secured and good safety messages prioritization scheme for cooperative awareness applications in VANETs. To achieve this, we propose a hybrid vehicle status-based and channel-aware-based BSM verification approach. These are:

- We proposed an algorithm to classify BSM into different safety areas (i.e. to reduce security problems) using received signal strength and
- We proposed an algorithm to rank BSM within their corresponding safety areas using transmitter-receiver distance and BSM arrival time (i.e. to improve awareness accuracy among neighboring vehicles).

## 6.3. Future works

All in all, our proposed scheme can address the excess receiving safety messages at vehicles in VANETs in dense traffic environments and achieve high awareness for neighboring vehicles with a high rate of relevant verified safety messages from nearby vehicles with low delay plus the security issues during BSM prioritization. So it can be a good candidate for a message prioritization scheme for cooperative driving safety applications.

Furthermore, our work can be extended in different ways.

- An obvious extension of the work could be to extend the algorithm for ranking more complex road scenarios. One can use characteristics of the driver (e.g. age, reaction time) combine with camera vision to increase the accuracy of verifying relevant safety messages.
- Another way could increase the performance of our proposed work by using a trust and reputation scheme to accept safety messages of trusted vehicles without verification.

## References

[1] WHO. (2015). Global status report on road safety 2015, "http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/" (Access Date: 19 May 2018).

[2] DOT HS 812 115. (2015). Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey.

[3] Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., & T., W. (2011). Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards, and Solutions. IEEE Communications Surveys & Tutorials, 13(4), 584 - 616.

[4] ETSI. (2013). ETSI - TR 101 607, Intelligent Transport Systems (ITS); Cooperative ITS (C-ITS); V1.1.1

[5] IVTS. (2014). IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture IEEE Std 1609.0-2013 (pp. 1-78). New York, NY 10016-5997, USA IEEE.

[6] Mishra, B., Nayak, P., Behera, S., & Jena, D. Security in Vehicular Ad hoc Networks: a Survey. Paper presented at the Proceedings of the International Conference on Communication, Computing and Security, New York, NY, USA. (2011)

[7] Raya, M., & Hubaux, J. P. (2007). Securing Vehicular Ad Hoc Networks. Journal of ComS. Biswas and J. Misic, “A cross-layer approach to privacy-preserving authentication in WAVE-enabled VANETs, ”IEEE Transactions on Vehicular Technology, vol. 62, no. 5, pp. 2182–2192, 2013.

[8] S. Biswas and J. Misic, “Relevance-based verification of VANET safety messages,” in Proceedings of the International Conference on Communications, pp.5124–5128, Xi’an, China, January 2012.

[9] J. H. Cheon and J. H. Yi, “Fast batch verification of multiple signatures,” in Proceedings of the 10th International Conference on Practice and eory in Public-Key Cryptography, pp. 442–457, Beijing, China, April 2007.

[10] E. B. Hamida, M. Javed, and W. Znaidi, "Channel-aware ECDSA signature verification of basic safety messages with K-means clustering in VANETs," Space-Based and Situated Computing, vol. 1, 2016.

[11] S. Biswas and J. Misic, "A cross-layer approach to privacy preserving authentication in WAVE-enabled VANETs," IEEE Transactions on Vehicular Technology, vol. 62, no. 5, pp. 2182–2192, 2013

[12] S. Banani, S. Gordon, S. Thiemjarus, and K. Somsak, "Verifying safety messages using relative-time and zone priority in vehicular Ad hoc networks," Sensors, vol. 18, 2018.

[13] S. Banani, S. Gordon, S. Thiemjarus, and K. Somsak, "Safety Message Verification Using History-Based Relative-Time Zone Priority Scheme," Hindawi, vol. 18, 2019.

[14] Petit, J. (2009, 2009/december). Analysis of ECDSA Authentication Processing in VANETs. Paper presented at the Proceedings of the 3rd International Conference on New Technologies, Mobility and Security.

[15] Petit, J., & Mammeri, Z. (2013). Authentication and Consensus Overhead in Vehicular Ad Hoc Networks. Telecommunication Systems, 52(4), 2699-2712.

[16] J. Issac and M. Shafie, "VANETs AND ITS RELATED ISSUES: AN EXTENSIVE SURVEY", Journal of Theoretical and Applied Information Technology, 1992-8645, 20th August, 2014.

[17] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang and Y. Zhou, "Heterogeneous Vehicular Networking: A Survey on Architecture, Challenges, and Solutions", IEEE Communications Surveys & Tutorials, 10.1109/COMST.2015.2440103, 2015.

[18] Rakesh Kumar1 and Mayank Dave "A Review of Various VANET Data Dissemination Protocols" International Journal of u- and e-Service, Science, and Technology Vol. 5, No. 3, September 2012

[19] IEEE. (2010a). IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation. IEEE 1609.4/D8.0, June 2010, 1-92.

[20] FCC. Dedicated Short-Range Communications (DSRC) Service, 2003, Available from: https://www.fcc.gov/wireless/bureau-divisions/mobility-division/dedicatedshort-range-communications-dsrc-service, (accessed on: 14 May 2020).

[21] IEEE. (2010b). Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments. IEEE Std 802.11p.

[22] Kenney, J. B. (2011). Dedicated Short-Range Communications (DSRC) Standards in the United States. Proceedings of the IEEE, 99(7). doi:10.1109/JPROC.2011.2132790

[23] IEEE. (2005). Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. IEEE Std 802.11e.

[24] Han, C., Dianati, M., Tafazolli, R., Kernchen, R., & Shen, X. (2012). Analytical Study of the IEEE 802.11p MAC Sublayer in Vehicular Networks. IEEE Transactions on Intelligent Transportation Systems, 13(2), 873-886.

[25] IEEE. (1998). IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements; Part 2: Logical Link Control, IEEE Std. 802.2, Edition (R2003), Also adopted by the ISO/IEC and re-designated as ISO/IEC 8802-2:1998.

[26] IEEE. (2002). IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture, IEEE Std. 802-2001. IEEE. (2005). Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. IEEE Std 802.11e.

[27] IEEE. (2016). IEEE Standard for Wireless Access in Vehicular Environments (WAVE)-Networking Services. IEEE Std 1609.3-2016 (Revision of IEEE Std 1609.3-2010), 1-160.

[28] SAE. (2009). Dedicated Short-Range Communications (DSRC) Message Set Dictionary, SAE Std.J2735 Int.

[29] Li, Y. (2012). An Overview of the DSRC/WAVE Technology. Paper presented at the Heterogeneous Networking for Quality, Berlin, Heidelberg.

[30] Cunha, F., Villas, L., Boukerche, A., Maia, G., Viana, A., Mini, R. A. F., & Loureiro, A. A. F. (2016). Data Communication in VANETs: Survey, Applications, and Challenges. Ad Hoc Networks, 44, 90 -103.

[31] ETSI. (2009). ETSI TR 102 638 V1.1.1, Vehicular Communications; Basic Set of Applications; Definitions.

[32] SAFESPOT. (2010). Cooperative Systems for Road Safety, Safespot Integrated Project - IST-4-026963-IP deliverable, Final report D8.1.1.

[33] C2C-CC. (2007). Car to Car Communication Consortium Manifesto: Overview of the C2C-CC System, Car to Car Communication Consortium, Tech. Rep. Version 1.1.

[34] ASTM E2213-03. (2010). 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.

[35] SAE. (2010). DSRC Implementation Guide, A guide to users of SAE J2735 message sets over DSRC. Retrieved from https://www.sae.org/standards/ (accessed on January 2018)

[36] NHTSA. (2014). DOT HS 812 014, "Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application".

[37] NHTSA. (2005). The CAMP Vehicle Safety Communications Consortium, Vehicle Safety Communications Project, Task 3 Final Report, Identify Intelligent Vehicle Safety Applications Enabled by DSRC, Tech. Rep. DOT HS 809 859. Retrieved from US Department of Transportation, Washington:

[38] Rakesh Kumar1 and Mayank Dave "A Review of Various VANET Data Dissemination Protocols" International Journal of u- and e-Service, Science, and Technology Vol. 5, No. 3, September 2012

[39] Md. Humayun Kabir, "Research Issues on Vehicular Ad hoc Network", International Journal of Engineering Trends and Technology (IJETT) – Volume 6 Number 4- Dec 2013.

[40] M. V. Eenennaam, L. Hendriks, G. Karagiannis, and G. Heijenk, "Oldest packet drop (OPD): a buffering mechanism for beaconing in IEEE 802.11p VANETs (poster)," in Proceedings of the Vehicular Networking Conference, pp. 252–259, Amsterdam, Netherlands, 2011.

[41] N.Gupta, A.Prakash, and R.Tripathi, "Adaptivebeaconingin mobility aware clustering-based MAC protocol for safety message dissemination in VANET," Wireless Communications and Mobile Computing, vol. 2017, 2017.

[42] Schmidt, R., Leinmuller, T., Schoch, E., Kargl, F., & Schafer, G. (2010). Exploration of Adaptive Beaconing for Efficient Intervehicle Safety Communication. Network, IEEE, 24(1), 14-19. doi:10.1109/MNET.2010.5395778

[43] C. Chigan and J. Li, "A delay-bounded dynamic interactive power control algorithm for VANETs," in Proceedings of the International Conference on Communications, pp.5849–5855, Glasgow, Scotland, UK, June 2007.

[44] H.Lua nd C.Poellabauer, "Balancing broadcast reliability and transmission range in VANETs," in Proceedings of the Vehicular Networking Conference, pp.247–254, Jersey City, New Jersey, USA, December 2010.

[45] K. A. Hafeez, L. Zhao, Z. Liao, and B. N. W. Ma, "A new broadcast protocol for vehicular Ad hoc networks safety applications," in Proceedings of the Global Telecommunications Conference, pp. 1–5, Miami, FL, USA, December 2010.

[46] M.V.Eenennaam, G.Karagiannis, and G.Heijenk, "Towards scalable beaconing in VANETs," in Proceedings of the Fourth ERCIM Workshop on eMobility, pp. 103–108, 2010.

[47] R. Stanica, E. Chaput, and A. L. Beylot, "Broadcast communication in vehicular ad-hoc network safety applications," in Proceedings of the Consumer Communications and Networking Conference, pp. 462–466, Las Vegas, NV, USA, January 2011.

[48] D.B.Rawat, D.C.Popescu, G.Yan, and S.Olariu, "Enhancing VANET performance by joint adaptation of transmission power and contention window size," IEEE Transactions on Parallel and Distributed Systems, vol.22, no.9, pp.1528–1535, 2011.

[49] D. B. Rawat, G. Yan, D. C. Popescu, M. C. Weigle, and S. Olariu, "Dynamic adaptation of joint transmission power and contention window in VANET," in Proceedings of the Vehicular Technology Conference Fall, pp. 1–5, Anchorage, AK, USA, September 2009.

[50] N. M. Mittal and S. Choudhary, "Comparative Study of Simulators for Vehicular Ad-hoc Networks (VANETs)", International Journal of Emerging Technology and Advanced Engineering, Volume 4, Issue 4, April, 2014.

[51] M. Boban and T. V. T. Vinhoza, "Modeling and Simulation of Vehicular Networks: Towards Realistic and Efficient Models", intechopen, USA and Portugal.

[52] Y. R. B. Al-Mayouf, M. Ismail, N. F. Abdullah, S. M. Al-Qaraawi and O. A. Mahdi, "survey on VANETs technologies and simulation models", ARPN Journal of Engineering and Applied Sciences, Volume 11, No. 15, August, 2016.

[53] The Network Simulator – ns-3. http://www.nsnam.org/. (Access Date: 19 June 2020).

[54] Simulation of Urban MObility (SUMO). http://sumo.sourceforge.net. (Access Date: 05 Feb 2020).

[55] Colliau, Taylor; Rogers, Grace; Hughes, Z.; and Ozgur, Ceyhun, "MatLab vs. Python vs. R" (2017). https://scholar.valpo.edu/cba_fac_pub/51

[56] https://realpython.com/matlab-vs-python (Access Date: 05 Nov 2020).

[57] Bu, Junling & Tan, Guozhen & Ding, Nan & Liu, Mingjian & Son, Caixia. (2014). Implementation and evaluation of WAVE 1609.4/802.11p in ns-3.

# Appendix A: Simulation Parameters of the BSM application in WAVE module on NS-3 simulator.

*a) Configurations of WAVE setup: wave_setup.cc*

```
#include "wave-setup.h"

namespace ns3
{

WaveSetup::WaveSetup(){}
WaveSetup::~WaveSetup () {}

NetDeviceContainer WaveSetup::ConfigureDevices (NodeContainer& nodes)
{
  /*
    Setting up WAVE devices. With PHY & MAC using default settings.
  */
  YansWifiChannelHelper waveChannel = YansWifiChannelHelper::Default ();
  YansWavePhyHelper wavePhy =  YansWavePhyHelper::Default ();
  wavePhy.SetChannel (waveChannel.Create ());
  wavePhy.SetPcapDataLinkType (WifiPhyHelper::DLT_IEEE802_11_RADIO);
  wavePhy.Set ("TxPowerStart", DoubleValue (5) );
  wavePhy.Set ("TxPowerEnd", DoubleValue (20) );
  wavePhy.Set ("TxPowerLevels", UintegerValue (8));
  //Setup up MAC
  QosWaveMacHelper waveMac = QosWaveMacHelper::Default ();
  WaveHelper waveHelper = WaveHelper::Default ();

  waveHelper.SetRemoteStationManager ("ns3::ConstantRateWifiManager",
                                       "DataMode", StringValue ("OfdmRate6MbpsBW10MHz" ),
                                       "ControlMode",StringValue ("OfdmRate6MbpsBW10MHz"),
                                       "NonUnicastMode", StringValue ("OfdmRate6MbpsBW10MHz"));

  NetDeviceContainer devices = waveHelper.Install (wavePhy, waveMac, nodes);
return devices;
}}
```

*b) Customizing of data_Tag to be broadcasted: my_data_tag.cc*

```
void MyDataTag::Serialize (TagBuffer i) const
{
        //we store timestamp first
        i.WriteDouble(m_timestamp.GetDouble());

        //then we store the position
        i.WriteDouble (m_currentPosition.x);
        i.WriteDouble (m_currentPosition.y);
        i.WriteDouble (m_currentPosition.z);

        //then we store the velocity
        i.WriteDouble (m_currentVelocity.x);
        i.WriteDouble (m_currentVelocity.y);
        i.WriteDouble (m_currentVelocity.z);

        //Then we store the node ID
        i.WriteU32(m_nodeId);
}
/** This function reads data from a buffer and store it in class's instance variables.
 */
void MyDataTag::Deserialize (TagBuffer i)
{
        //We extract what we stored first, so we extract the timestamp
        m_timestamp =  Time::FromDouble (i.ReadDouble(), Time::NS);;

        //Then the position
        m_currentPosition.x = i.ReadDouble();
        m_currentPosition.y = i.ReadDouble();
        m_currentPosition.z = i.ReadDouble();
```

### c) Configuration of BSM broadcasting function: NaolyThesisAp.cc

```
void
NaolyThesisAp::BroadcastInformation()
{

    NS_LOG_FUNCTION (this);
    //Setup transmission parameters
    TxInfo tx;
    tx.channelNumber = CCH;
    tx.priority = 7; //highest priority.
    tx.txPowerLevel = 7;
    tx.dataRate = WifiMode("OfdmRate6MbpsBW10MHz");

    Ptr<Packet> packet = Create <Packet> (m_packetSize);

    //let's attach our custom data tag to it
    MyDataTag tag;
    tag.SetNodeId ( GetNode()->GetId() );
    tag.SetPosition ( GetNode()->GetObject<MobilityModel>()->GetPosition());
    tag.SetVelocity ( GetNode()->GetObject<MobilityModel>()->GetVelocity());
    //timestamp is set in the default constructor of the MyDataTag class as Simulator::Now()

    //attach the tag to the packet
    packet->AddPacketTag (tag);

    // Broadcast the packet as WSMP (0x88dc) if the node is moving

        m_waveDevice->SendX (packet, Mac48Address::GetBroadcast(), 0x88dc, tx);

    //We will still schedule the next broadcast, in case the node started moving again
    Simulator::Schedule (m_broadcast_time, &NaolyThesisAp::BroadcastInformation, this);
}
```

### d) Configuration of BSM receiving function: NaolyThesisAp.cc

```
bool
NaolyThesisAp::ReceivePacket (Ptr<NetDevice> device, Ptr<const Packet> packet,uint16_t protocol, const Address &sender)
{
    NS_LOG_FUNCTION (device << packet << protocol << sender);
    /*
        Packets received here only have Application data, no WifiMacHeader.
        We created packets with 200 bytes payload, so we'll get 200 bytes of payload.
    */

    NS_LOG_UNCOND ("\nNode " << GetNode()->GetId() << " : Received BSM from " << sender << " Size:" << packet->GetSize());

    //Let's check if packet has a tag attached!
    MyDataTag tag;
    if (packet->PeekPacketTag (tag))
    {


      m_outFile
       <<tag.GetTimestamp().GetMilliSeconds()<<","
       <<GetNode()->GetId()<<","
       <<tag.GetNodeId()<<","
       <<tag.GetPosition().x<<","<< tag.GetPosition().y<<","
       <<Now()<<std::endl;
      m_outFile.flush ();
    }

    return true;
}
```

## Appendix B: Simulation of analysis on received BSM a given vehicles on Jupyter Notebook.

*i) Importing important python libraries:*

In [1]:
```
#.......................................................................................................
#  author : Naol Getachew
#  date :   12/10/2020
#.......................................................................................................

#...to import neccessary librabries...
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
from sklearn.cluster import KMeans
```

*ii) Load received BSM by vehicles: Let's look at for Node-3.csv*

In [12]:
```
#Sample display the data received by Node-3 at time T(ms)
BSM_data = BSM_data[BSM_data['Brdcast_Time(ms)'] == 186000]
print("The BSM Node-3 received from its Neighboring Nodes are:\n", BSM_data)
```

```
The BSM Node-3 received from its Neighboring Nodes are:
       Brdcast_Time(ms)  Send_Id       Arrival_Time  Received_Power   distance
2768            186000       63  +186101917051.0ns        -34.9857   5.339383
2769            186000       33  +186202977291.0ns        -47.2256  13.397500
2770            186000       16  +186203403310.0ns        -45.3147  11.557364
2771            186000        2  +186403686278.0ns        -28.8322   3.467500
2772            186000       77  +186703221309.0ns        -41.8635   8.948229
2773            186000       31  +186801543050.0ns        -38.5764   6.994859
2774            186000       52  +186904375317.0ns        -47.0142  13.177398
```

*iii) Assign the BSM as Data-Frame data for further experiments*

In [4]:
```
#Note: we used the fixed cluster centroid values from the experiment done by existing work [10]!
#.........................................NaolyRankingScheme.............................................

#Load bsm data to dataframe
df = pd.DataFrame(BSM_data)
df = df.round(3)
```

In [5]:
```
#Create a list of all rss in the 'df'
list_of_rss = []
for i in range (-90, -30):
    list_of_rss.append(i)
```

*iv) Sample of classification and ranking for BSM assigned to the first safety area*

In [13]:
```
# breif: We used rss value from experiment on 'accuracy of BSM classification based on K-Means Clustering' by in [10]
#.......................................................................................................
print("...................................................................")

#.....the 1st safety area.....
for rss in list_of_rss:
    condition = df['Received_Power'] > -62
    SA1 = df[condition].dropna()

#If the filtered data has some records
print("\nBSMs found between distance of 0-50m are:")
if len (SA1) > 0:
    #Now Rank BSMs within SA1 Using distance values
    SA1["New_Rank_SA1"] = SA1['distance'].rank()
    SA1.sort_values('distance', inplace = True)

    #Now Rank BSMs within SA1 Using arrival values
    SA1["Old_Rank_SA1"] = SA1['Arrival_Time'].rank()
    SA1.sort_values('Arrival_Time', inplace = False)

    #After ranking
    print("** SA1 After Rank **")
    print(SA1)
```

End!

Glory to my God!