MASTER

Design and evaluation of multi-channel operation implementation of ETSI GeoNetworking Protocol for ITS-G5

Priandono, R.

Award date:
2015

Link to publication

Master Thesis

Rangga Priandono

Graduation Supervisor:
Prof. Dr. J.J. Lukkien (TU/e)

Graduation Tutors:
Dr. Ir. Jan De Jongh (TNO)
Ir. J.Van Der Sluis (TNO)

Final version

Eindhoven, August 2015
Abstract

The Intelligent Transport System goal is to make the transportation system more robust (e.g. safe, secure, etc.). One of the methods allows the vehicles to communicate to each other. The ETSI (Europe) and the IEEE (US) are two organizations that create the standard model for the Inter-Vehicular Communication, which are the ETSI-ITS model and the WAVE model respectively. Although the Multi-Channel Operation is supported, the ETSI standard document does not contain any part that explains about the channel assignment mechanism to implement the Multi-Channel Operation (MCO) on the ITS-G5 frequency.

This thesis contributes three things: 1) Build a VANET ETSI-ITS simulation model using the NS3 library as a tool. 2) Define five strategies and compare them through simulation. 3) Propose a new metric (i.e. accuracy) that can be used for a more practical approach. Moreover, these five strategies are tested using two scenarios which are the single domain and the multi domain scenario, and the results are measured using three metrics (i.e. reliability, accuracy, and effectiveness).

The VANET ETSI-ITS simulation model simulates all the possible conditions in the ETSI-ITS model. These possible conditions are the 802.11p broadcast back off procedure, the channel-load information sharing, the ETSI-ITS cross layer architecture, propagation model, etc. The results showed one strategy came as the best strategy because it shows the best performance in most metrics. Meanwhile, two strategies present a unique behavior that can be used by a specific user.
4 Simulation Results 29
  4.1 The variation simulation result 29
  4.2 Channel Monitor Decision Interval 30
  4.3 Single Domain Scenario 32
    4.3.1 Success Message Ratio 32
    4.3.2 Message Drop 33
    4.3.3 Information Age 34
  4.4 Multi Domain Scenario 35
    4.4.1 Success Message Ratio 35
    4.4.2 Message Drop 37
    4.4.3 Information Age 38
  4.5 Discussion 39

5 Conclusions and Future Works 43
  5.1 Summary 43
  5.2 Conclusion 44
  5.3 Future Works 45

Bibliography 47

Appendix 51

A Model Validation 51

B CSMA/CA Broadcast Procedure 56

C Transmit and Receive Message Source Code 58
  C.1 Transmit message source code 58
  C.2 Receive message source code 62
List of Figures

1.1 Maximum limit of mean spectral power density for each channels [13] .......... 3  
1.2 The ETSI ITS model and the ITS-station architecture ............................. 5  
3.1 Comparison between WAVE and ETSI model according to OSI layers.[1] ........ 12  
3.2 ETSI ITS-Station reference architecture [9] ........................................... 13  
3.3 Hidden nodes a.k.a Hidden terminal .................................................... 20  
3.4 vehicles position in simulation model scenarios ...................................... 20  
3.5 Sharing Local CBR information algorithm [13] ....................................... 22  
4.1 Box plot anatomy ................................................................................. 29  
4.2 The LCBR strategy channel load between 10 to 50 ms from 30 simulation runs .. 30  
4.3 The LCBR strategy Channel load between 4 to 20 ms from 30 simulation runs .. 31  
4.4 The average SMR in single domain scenario .......................................... 32  
4.5 The fairness from the top 3 strategies in single domain scenario ................ 33  
4.6 The average MD in single domain scenario .......................................... 34  
4.7 The average IA in single domain scenario .......................................... 34  
4.8 The average SMR in multi domain scenario .......................................... 35  
4.9 The average local CBR in the LCBR strategy from every vehicles in multi domain scenario ................................................................. 36  
4.10 The fairness from the LCBR and RDM strategy in Multi domain scenario .... 37  
4.11 The average MD in multi domain scenario .......................................... 37  
4.12 The average IA in multi domain scenario .......................................... 38  
4.13 The average CW from every vehicles in multi domain scenario ............... 39  
A.1 SCH1 ............................................................................................... 51  
A.2 SCH2 ............................................................................................... 51  
A.3 CCH ................................................................................................. 51  
A.4 Validation topology ............................................................................. 52  
A.5 Parallel transmit scenario ..................................................................... 52  
A.6 Message drop scenario ........................................................................ 52  
A.7 Message out of range scenario. ............................................................. 53  
A.8 Message out of range scenario in detail. .............................................. 53  
A.9 one-hop behavior scenario topology .................................................... 54  
A.10 one-hop behavior scenario .................................................................. 54  
A.11 Hidden node scenario ........................................................................ 54
LIST OF FIGURES

A.12 Message size validation ........................................ 55
A.13 CBR dissemination topology .................................. 55
A.14 CBR dissemination scenario .................................. 55

B.1 CSMA/CA Broadcast mode procedure [2] .................... 56
List of Tables

3.1 The 802.11p parameter settings [3] [13] [24]. ......................... 15
3.2 The simulation parameter settings [3]. ................................. 19
3.3 The summary list of test. .................................................. 28

4.1 The summary of pro and cons from each strategy in single domain scenario. .... 40
4.2 The summary of pro and cons from each strategy in multi domain scenario. .... 40
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITS</td>
<td>Intelligent Transport System</td>
</tr>
<tr>
<td>WAVE</td>
<td>Wireless Access in Vehicular Environments</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunications Standards Institute</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>C-ITS</td>
<td>Cooperative-Intelligent Transport System</td>
</tr>
<tr>
<td>DCC</td>
<td>Decentralize Congestion Control</td>
</tr>
<tr>
<td>GN</td>
<td>GeoNetworking</td>
</tr>
<tr>
<td>VANET</td>
<td>Vehicle Ad-hoc Network</td>
</tr>
<tr>
<td>MANET</td>
<td>Mobile Area Network</td>
</tr>
<tr>
<td>WMN</td>
<td>Wireless mesh Network</td>
</tr>
<tr>
<td>MCO</td>
<td>Multi Channel Operation</td>
</tr>
<tr>
<td>MCMI</td>
<td>Multi Channel Multi Interface</td>
</tr>
<tr>
<td>ITS-G5</td>
<td>name of a frequency that dedicated for ITS</td>
</tr>
<tr>
<td>CCH</td>
<td>Control Channel</td>
</tr>
<tr>
<td>SCH</td>
<td>Service Channel</td>
</tr>
<tr>
<td>BTP</td>
<td>Basic Transport Protocol</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>SHB</td>
<td>Single Hop Broadcast</td>
</tr>
<tr>
<td>TSB</td>
<td>Topology Scope Broadcast</td>
</tr>
<tr>
<td>LS</td>
<td>Location Service</td>
</tr>
<tr>
<td>RSU</td>
<td>Road-Side Unit</td>
</tr>
<tr>
<td>BSM</td>
<td>Basic Safety Message</td>
</tr>
<tr>
<td>CAM</td>
<td>Cooperative Awareness Message</td>
</tr>
<tr>
<td>DENM</td>
<td>Decentralized Environmental Notification Message</td>
</tr>
<tr>
<td>NDL</td>
<td>Network Design Limits</td>
</tr>
<tr>
<td>STF</td>
<td>Special Task Force</td>
</tr>
<tr>
<td>ITS-S</td>
<td>Intelligent Transport System - Station</td>
</tr>
<tr>
<td>CEN DSRC</td>
<td>European Committee Standardization for Dedicated Short Range Communication</td>
</tr>
<tr>
<td>CBR</td>
<td>Channel Busy Ratio</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal frequency-division multiplexing</td>
</tr>
<tr>
<td>IVC</td>
<td>Inter-Vehicular Communication</td>
</tr>
<tr>
<td>HGS-TS</td>
<td>Harmonize Group Special - Task Force</td>
</tr>
</tbody>
</table>
LIST OF TABLES

TNO ............ Netherlands Organisation for Applied Scientific Research
SAM ............ Service Announcement Message
NS ............ Network Simulator
DCF ............ Distributed Coordination Function
CSMA/CA ...... Carrier Sense Multiple Access / Collision Avoid
SINR .......... Signal Noise Ratio
AIFS .......... Arbitrary Inter-Frame Spacing
CW ............ Contention Window
RTS .......... Request To Send
CTS .......... Clear To Send
LocT .......... Location Table
MAC .......... Media Access Control
PHY .......... Physical
OSI .......... Open Systems Interconnection
LLC .......... Logical Link Control
UDP .......... User Datagram Protocol
oPBC .......... one Hop Periodic Communication
CR .......... Communication Range
OOR .......... Out Of Range
SMR .......... Success Message Ratio
MD .......... Message Drop
IA .......... Information Age
LCBR .......... Local CBR strategy
GCBR .......... Global SBR strategy
LD .......... Least Delay Strategy
RDM .......... Random Strategy
RBN .......... Round Robin Strategy
Chapter 1

Introduction

1.1 Intelligent Transport System

People use transport systems to travel from one place to another. People try to improve these transportation systems to be more sustainable, meaning more efficient, safe, clean, and seamless. The Intelligent Transport System (ITS) is a concept that works towards this definition of sustainability.

The European Telecommunication Standard Institute (ETSI) is an organization that responsible to design and standardize the ITS technologies. The Harmonize Group Special Task Force (HGS-TS) is a group of people from Europe and the US who are responsible evaluate the ITS technologies. TNO, which is the Netherlands research institution, actives in the ETSI ITS technology because their network technology department is a member of the ETSI HGS-TS.

The ITS technologies can be applied to any transportation systems (i.e. road, air, water, and rail) and they are greatly dependent on computers, electronics, satellites and/or sensors. An example is the eCall (emergency call) service that automatically calls emergency services and transmits the location of the accident scene when a vehicle experiences an accident. The eCall service reduces the emergency service response time and minimizes the number of severe injuries by 5-10% [4].

1.2 GeoNetworking Protocol

The ITS technologies will work properly when all vehicles are able to communicate; therefore, a protocol is needed to provide a communication feature for all vehicles. Many protocols had been made by the automotive industries to provide communication between vehicles before the GN protocol was published; however, because there was no standard protocol, these vehicles could only communicate with other vehicles that were manufactured by the same company [18].

The ETSI committee determines the GeoNetworking (GN) protocol as the standard protocol for inter-vehicular communication in Europe. The committee published two standard documents about how to implement the GN protocol based on the media access frequencies. First is the media-independent functionalities document (ETSI EN 302 636-4-1), which describes the GN specification when it is implemented on common ITS access technologies (e.g. Bluetooth, infrared, Wi-Fi, 2G/3G, etc.). The second is the Media-dependent functionalities document (ETSI TS 102
636-4-2), which describes the GN specification when it is implemented on the ITS-G5 media access (i.e. media access devices that work on the ITS-G5 frequency).

These standard documents still have missing parts. For example, in the media-dependent document, there is a recommendation to use the multi-channel operation when implementing GN on ITS-G5. However, there is no part in the document that describes the channel assignment strategy for multi-channel multi-transceiver operations. The channel assignment strategy is expected to describe how to use the available channels in order to increase the network capacity.

This thesis focuses on the second document because the ITS-G5 is the designated frequency for intelligent transport systems in Europe [8]. This thesis contributes three things: 1) Build a VANET ETSI-ITS simulation model using the NS3 library as a tool. 2) Define five strategies and compare them through simulation. 3) Propose a new metric (i.e. accuracy) that can be used for a more practical approach. Moreover, the five strategies are tested using two scenarios which are the single domain and the multi domain scenario, and the results are measured using three metrics (i.e. reliability, accuracy, and effectiveness).

1.2.1 GeoNetworking Protocol Media-Dependent

The GN media-dependent standard is an important document because it guides engineers to implement the GN protocol on ITS-G5 media access. The document covers information about the ITS-G5 frequency, the type of safety-road messages with the extra information carried by the message, the implementation of Multi-Channel Operation (MCO), the interference mitigation techniques between CEN DSRC and Cooperative ITS (C-ITS), and the Decentralized Congestion Control (DCC) for channel load information sharing strategy [13].

The ITS-G5 frequency is divided into three subclasses, which are the ITS-G5A for safety-road applications, the ITS-G5B for non-safety road applications, and the ITS-G5D for future applications. Figure 1.1 depicts the ITS-G5 frequency band and the maximum allowed output power level on each channel. The ITS-G5 frequency band is divided into seven channels with 10 MHz bandwidth per channel, these channels are the Control channel (CCH) and six Service channels (i.e. SCH1 to SCH6) [13][11].

On the ETSI ITS-G5 there are two types of safety-road messages, the cooperative awareness message (CAM) that contains vehicles basic status information (i.e. position, heading, speed, etc.) and the warning message (DENM) that contains road hazard information; these messages are disseminated using GN SHB packet that has single hop behavior and GN GeoBroadcast packet that has multi-hop behavior respectively. Furthermore, the CAM message is also used to carry extra information about neighbor vehicles channel load. The extra information is stored in a Location Table (LocT). The LocT is used to hold information about other vehicles that are also using the GN protocol[13].

The decision for using the MCO is meant to increase the network capacity by utilizing all the available channels. The MCO can be implemented using single transceiver or multi transceivers. For safety-related context in the single transceiver, the transceiver should be tuned on the control channel; hence, there is no channel assignment mechanism needed. Moreover, there is no technical description for non-safety related context using a single transceiver; this missing technical description can be included in future works.

Meanwhile, for implementing MCO using the multi transceivers technique, the document proposes two transceivers per vehicle. The first transceiver acts as fixed interface that only tunes
to the Control channel and the second transceiver acts as a switchable interface that can switch between the Service channels. For safety-related context in multi transceivers, vehicles can broadcast messages from the control channel (via fixed interface) or the service channels (via switchable interface). For non-safety-related context, vehicles may use the Service Announcement Message (SAM) that is broadcasted via fixed interface to synchronize the switchable interface on both sender and receiver vehicles [13].

Furthermore, the media-dependent document describes the mitigation techniques to avoid interference between the ITS-G5 with the CEN DSRC that is used for electronic toll collection. The first technique is to reduce the transmit power when a vehicle nears a toll station. The second technique is to adjust the packet interval based on the number of surrounding vehicles. The last technique is to adjust the output power based on the transmit rate. However, these mitigation techniques are not needed if the ITS-G5 onboard unit is installed 1.5m away from the CEN DSRC onboard unit and the output power of the ITS-G5 is set to max 10 dBm [13].

The goal of sharing the DCC channel load information is to reduce the number of message collisions caused by the hidden terminals in the wireless network. A vehicle knows about the channel load condition of its 1-hop and 2-hops neighbor vehicles by sharing the DCC channel load information. The channel load information is represented and calculated using the Channel Busy Ratio (CBR). The CBR is a ratio that represents the channel load on a vehicle, the ratio denotes the number of incoming/outgoing messages in predetermined time interval; therefore, every vehicle may have a different ratio, and the predetermined time interval plays an important role to determine how busy a channel is [13].

In order to measure the proposed channel assignment strategy, this thesis builds a simulation model based on the description of the media-dependent standard document. The simulation model is designed to be able to simulate arbitrary vehicle configurations. The simulation model simulates the ITS-G5A subclass frequency because it is designed for road-safety applications. Furthermore, the CAM message is chosen for the simulation model because it is also used to disseminate the channel load information [13]. Hence, the simulation model simulates the GN SHB packet behavior, which is a single hop behavior [14]. The simulation model focuses on

---

**Figure 1.1:** Maximum limit of mean spectral power density for each channels [13]
CHAPTER 1. INTRODUCTION

the multi transceivers technique because there is no channel assignment mechanism in the single transceiver for safety-related context. The mitigation technique to avoid interference between the ITS with the CEN DSRC is not simulated because we assume the vehicle does not install the CEN DSRC onboard unit. The model also simulates the CBR information dissemination for the channel assignment strategy that is based on the CBR value. Additionally, to eradicate the possibility of adjacent channel interference in the simulation model, all transceivers are assumed to have a good spectral mask and sets with an equal amount of transmit power [13].

1.3 Problem Description

The media-dependent standard document recommending the MCO either with the single or the multi transceivers technique when implementing the GN protocol on ITS-G5. However, there is no part in the standard document describes the channel assignment strategy for multi-channel multi-transceiver operations. This condition brings us to the main question.

The main question is:

*How can we determine the proper channel for a packet in ITS-G5 Multi Channel Operation?*

1.4 Research Question

The main challenge is to design the best strategy and evaluates the use of MCO in ETSI GeoNetworking. The main idea is to add information of a proper channel for an outgoing message in layer-3. According to the main question, the following research questions are defined:

- **What are the possible strategies to define the proper channel for a packet in ETSI GeoNetworking Multi-Channel Operations?**
  For instance, which information can be used for channel assignment strategy to reduce the single-channel load during heavy-traffic situations and prevent other channels from becoming useless.

- **What is the expected performance gain of Multi-Channel Operations?**
  This can be seen by observing the reliability, the accuracy, and the effectiveness properties of the vehicles when implementing a particular channel assignment strategy.

1.5 Approach

1.5.1 Related works

The literature review focuses on implementing the ETSI GN protocol Multi-channel Operation on ITS-G5. The goal is to find other strategies of implementing the multi-channel multi-transceiver on ITS-G5 that have been proposed by other people before this thesis started. Moreover, the literature review also considers the possible advantages and drawbacks when using a particular strategy that was proposed by other people.
CHAPTER 1. INTRODUCTION

1.5.2 Model design

The simulation model is built to represent an environment to test the proposed channel assignment strategy. In general, the simulation model simulates a Vehicular Ad-hoc Network (VANET) environment based on the ETSI standard document. Figure 1.2 (a) depicts the mapping of the VANET ETSI model layers onto the OSI layers. Furthermore, every vehicle in the model is built based on the ETSI ITS-Station architecture reference, which is shown in figure 1.2 (b). The model simulates the highway traffic scenarios during busy hours. Additionally, the vehicle model behavior simulates one hop periodic broadcast communication because it simulates the CAM message that is broadcast using the GN SHB packet. The GN protocol contains 6 modules, which are the GeoUnicast, the GeoBroadcast, the Beacon, the GeoAnycast, the Topology Scope Broadcast, and the Location Service. the Single-Hop Broadcast is a sub-module of the GN Topology Scope Broadcast module that have the one-hop behavior.

1.5.3 Simulation tool

A simulation tool is needed to build the simulation model. Nowadays, there are many types of computer simulation tools that can be used to run a simulation model. However, a network simulation tool is the proper tool for this particular model because it can be used to simulate network communication and to evaluate the network communication performance on the physical, data link, network layer, and transport layer.

There are eight network simulation tools that can be used to simulate the model, these tools are NS2, NS3, OPNET, NETSIM, OMNET++, JSIM, QUALNET, and REAL [23]. To shorten the list, two requirements need to be fulfilled by the network simulation tools. First, the tool should be open source. The first requirement drops all the commercial tools and leaves five tools on the list (i.e. NS2, NS3, J-Sim, OMNET++, and Real). Second, the tool should be up to date. The Real version 5.0 was released on 13th of August 1997, the NS2 version ns-2.35 was released on 4th of November 2011, the J-Sim version 1.3 latest patches was uploaded on 8th of May 2013, Omnet++ version 4.6 was uploaded on 12th of February 2014 and the NS3 version ns-3.22 was released on 5th of February 2015. Hence, the NS-3 network simulation tool is selected as the
CHAPTER 1. INTRODUCTION

A proper tool to simulate the MCO implementation on ITS-G5.

1.5.4 Result Evaluation

A quantitative evaluation method is chosen to measure the reliability, the accuracy and the effectiveness of channel assignment strategy. The method measures three properties, which are the successful message ratio of the sender vehicle for reliability, the average information age of successful messages for accuracy, and the average of message dropped on the sender vehicles for effectiveness. The successful message ratio is a proper metric to observe how effective the strategy to overcome the 802.11 broadcast mechanism flaws in an ad-hoc network. Moreover, the information age is an important metric because it calculates how old the information of sender vehicle position, which is located on the receiver vehicle location table, before it is updated by the new incoming message. The last metric is used to observe how effective the channel assignment strategies to reduce the number of message drops that is caused by a long delay.

The inter-vehicular communication is based on the IEEE 802.11 standard. To avoid collisions in an ad-hoc network, the 802.11 wireless families (including the 802.11p) use the Distributed Coordination Function (DCF) to manage the channel access. When using DCF, vehicles deploy the CSMA/CA that uses the RTS/CTS handshake and random back-off period. Furthermore, in point-to-point communication, the source first checks if the channel is clear or not. If the channel is clear, the source continues by sending an RTS (Request To Send) message to the destination. Once the destination receives the RTS message, it replies with a CTS (Clear To Send) message. Then, the source sends the information to the destination, and the process is ended by an acknowledgment that is sent by the destination. However, if the source did not receive a CTS from the destination or the channel is not clear, the source initiates a random back-off period and redoes all the steps. Meanwhile, the broadcast message does not use the RTS/CTS handshake and the acknowledgment because there is no destination. To send a broadcast message, the source only checks if the channel is clear or not. This broadcast mechanism flaw increases the collision problems (i.e. the hidden nodes and the contention problem) [2].

When using the GN protocol, vehicles use the Location Table to record the position of the neighbor vehicles. The Location Table is updated from the incoming messages (e.g. CAM) that are sent by the neighbor vehicles. Because of the limitation of a transceiver, which cannot transmit and receive messages at the same time, a CAM message that is ready to be transmitted should be queued when the transceiver is busy receiving messages. During a busy period, the queued CAM message can be put on delay state until it exceeds the next message activation time. This condition makes the information inside the queued CAM message no longer valid; hence, the queued CAM message needs to be dropped. Otherwise, the queued CAM message is safe to be transmitted. Additionally, the detailed descriptions of evaluation metrics are discussed in chapter three.

1.6 Outline

The rest of this thesis is structured as mention below,
chapter 2 describes the related works,
chapter 3 defines the methodology,
chapter 4 discusses the simulation results,
and chapter 5 is the conclusion and future works.
Chapter 2

Related Works

There have been meagre papers discussed the implementation strategy of GN protocol MCO on ITS-G5. Therefore, we expanded our scope to find papers that discuss the implementation of Multi-Channel Multi-Interface (MCMI) in other domains. The goal is to find any possible channel assignment strategies that can be used to implement the GN protocol MCO on ITS-G5. From the literature study, four papers are considered useful for this thesis. The first two papers support the number of transceivers that should be installed per vehicle while the last two papers propose other strategies that might applicable for the simulation model.

Kyasanur et al. (2005) discussed the advantage of multi-interface technique compared to the single interface technique in multi-channel operations. The advantage is determined by measuring the network capacity on every network configurations. On the paper, the notation \( M \) represents the number of interfaces and \( C \) denotes the number of channels, while notation \( W \) represents the data rate of a channel. The paper states that in arbitrary network, a network configuration where \( M = C \) has higher network capacity compared to the network configuration where \( M < C \). The authors model two types of channels, the first channel has a flexible data rate, which is defined by dividing the data rate by the number of channels \((W/C)\). The second channel has fixed data rate, which means the aggregate of the data rate equals to the data rate multiplied by the number of channels \((W*C)\). Furthermore, the authors also model two types of networks: 1) the arbitrary network, where all nodes location and the traffic patterns can be controlled, and 2) the random network, where all nodes location and the traffic patterns cannot be controlled. Additionally, the paper models the switching delay mechanism for the switchable interface. The paper also claims that in random networks, the \( M < C \) configuration has the same network capacity as the \( M = C \) configuration, as long as the upper bound ratio of \( C \) and \( M \) in the \( M < C \) configuration is \( O(\log n) \). Otherwise, the network capacity of the \( M < C \) configuration is lower than the \( M = C \)[19].

In their next paper, Kyasanur et al. (2006) validate their claim about the network capacity in random networks from the previous paper by using a simulation model. The node configurations on the simulation model illustrate the technique of multi transceiver implementation, which is stated on the media-dependent standard document. They create a Mobile Ad-hoc Network (MANET) model that simulates the condition of the random network. Each node on the simulation model had the \( M < C \) configuration and installed with two interfaces (transceivers). The first interface was configured to stay tuned on a single channel (i.e. fixed interface) and the other was designed to
be able to switch between channels (i.e. switchable interfaces). The model ensures the nodes fixed interface was used only for receiving messages and the switchable interface was only used to send messages or forwarding messages during the simulation. Therefore, every transmitted message was assigned to the channel ID of the destination node. For an illustration, node A, which is act as the destination node, has channel three as its channel ID; hence, the fixed interface on node A is set permanently on channel three. Moreover, on the sender nodes neighbor-table, node A was labeled with channel three. Therefore, every time a sender node wants to send a message to node A, its switchable interface needs to switch communication to channel three. However, the paper states that the simulation model can experience performance degradation from broadcast type messages due to switching channel on the switchable interface[20].

Raniwala et al. (2004) wrote another paper that discusses the implementation of multi-channel multi-interface. The authors evaluated the multi-channel operation in a multi-hop wireless ad-hoc network architecture model. The architecture model was designed based on the standard 802.11 hardware equipment. Every node in the model has multiple interfaces that can operate on different channels, and the model simulates a Wireless Mesh Network (WMN). For the channel assignment, the authors propose a mechanism based on channel traffic load information and combine it with two routing algorithms. The goal was to exploit the multiple channels in WMN to meet the need of a high capacity requirement to support the backbone capacity of the WMN. The authors design the WMN topology with several fixed wireless routers, which form a multi-channel Ad-hoc wireless network. One of the fixed routers has a wired connection and acts as the network gateway to the internet. Two of the fixed wireless routers on the topology cover the end-user mobile station. In the multi-hop Ad-hoc network, the wireless routers relay packets among themselves, from the internet to the end-user mobile station or vice versa. Moreover, each wireless router in the network uses two fixed interfaces. That means the number of channels that can be used by a router at the same time cannot be more than two. The channel assignment algorithm that uses the traffic load information is named the load-aware channel assignment. During the simulation, the load-aware channel assignment assigned packets to a channel that had the least traffic load, and the load-aware channel assignment was also combined with two types of routing algorithms, which are the short path routing and the randomize multi path routing. The result shows that the load-aware channel assignment yielded the full potential of multi-channel in WMN. The load-aware channel assignment algorithm was based on the traffic load information that measured the channels load. Unfortunately, the load-aware channel assignment algorithm is not designed to reduce the number of collisions problem that are caused by the hidden nodes and the contention problem[22].

Liu et al. (2010) discussed the potential of cross-layer architecture in the MCMI design for a real-time video streaming application. The cross-layer architecture was chosen because it is expected to enhance the wireless system performance (i.e. throughput and image quality). Furthermore, the cross-layer architecture allows some layers to use the information that exists on the other layers; hence, it would make a better strategy decision for the channel assignment mechanism. The authors select information from the physical, the data link, and the application layer of the cross-layer architecture. To measure the performance, the cross-layer channel assignment strategy was implemented using a simulation tool. For the model topology and node configuration, the authors decide to use the Kyasanur et al. (2006) model configuration, where the wireless network model was based on the IEEE 802.11 standard. Every node in the model has two types of interfaces which are the fixed and the switchable interface where the total number of channels are greater than the number of interfaces ($M < C$). The information of available bandwidth
on the MAC layer and the information of packet queue length on the transceivers were used for the channel assignment mechanism, and the system performance was analyzed by measuring the throughput and the packet loss ratio. The model simulated a spacious $1000 \times 1000$ square meters environment with 16 nodes that were placed in random positions. These random nodes simulated both mobile and static nodes. The results show that the MCMI using the cross-layer had a better throughput and a better image quality compared to the non-cross-layer MCMI. However, the simulation model used unicast packets, this may be caused by the Kyasanur et al. (2006) model that had performance degradation when using broadcast packets [21].

The simulation model in this thesis will use the $M = C$ configuration, which is the number of transceivers on each vehicle equals to the number of ITS-G5A channels, in order to reach a maximum network capacity in MCO. Furthermore, the $M = C$ configuration also avoids the model to use the switchable interface that may cause performance degradation when broadcasting the CAM messages using the GN SHB packet. In this thesis, two of the channel assignment strategies use the channel load information. The CBR is the recommended technique that is stated on the standard document for implementing the GN protocol on ITS-G5; therefore, this thesis will use the channel load information that is calculated based on the CBR. Moreover, this thesis will not use the transmit queue information as one of the channel assignment strategies, even though it is proposed by one of the papers and the ETSI DCC access can provide the transmit queue information. The reason for not using the transmit queue information is because the model has only max one message per period stored in queued. Due to the model behavior, the queued message that exceeds next interval will be dropped. More detail about the simulation model behavior is described in section 3.1.3.
Chapter 3

Methodology

This chapter describes the steps that are needed to build the simulation model. This chapter is divided into five sections which are the approach that is used to build the simulation model, the scenarios, the Channel Busy Ratio (CBR) which describes the channel load measurement, the proposed strategies for channel assignment mechanism, and the metrics to measure the strategies performance. The simulation model should be able to simulate a variety of Vehicles Ad-hoc Network (VANET) scenario environments, in order to test the proposed channel assignment strategies.

Currently, there are two Inter-Vehicular Communication (IVC) architecture models. First, the WAVE model that follows the IEEE standard on the data link and the network and transport layer (i.e. the IEEE 1609.3 and the IEEE 1609.4). Second, the ETSI ITS model that follows the ETSI standard on the data link and the network and transport layer (i.e. the Decentralize Congestion Control, the GeoNetworking protocol and the Basic Transport Protocol). Furthermore, on the physical layer, both architecture models operate using the IEEE 802.11p standard. Figure 3.1 shows the comparison between the WAVE model and the ETSI ITS model according to the OSI layers.

The DCC is a component that distinguishes between the ETSI model and the WAVE model, and it provides information from a particular layer. For example, the DCC function that is located in the access layer provides information about the data link and the physical layer (e.g. Channel load, transmit queue, PHY status, etc.). According to the ITS-Station reference architecture, the DCC functions are spread in four different layers which are the DCC-access on the access layer, the DCC-network on the network and transport layer, the DCC-application on the facilities layer, and the DCC-management on the management cross-layer [10].

The main objective of the DCC functions is to provide information that can be used as a reference to reducing the number of message collisions. One of the techniques that use the DCC function is the transmit power adjustment control [25]. Another technique that uses the DCC function is the message transmit rate control [24]. Both techniques have a purpose to reduce the number of messages in a communication area; hence, these two techniques can reduce the number of message collisions.

The simulation model simulates the ETSI DCC function and uses it as an input for two of the channel assignment strategies. In the ETSI model, the DCC function plays an important role because it provides information to reduce the number of message collisions. There are three DCC functions that need to be simulated which are the DCC-access, DCC-network, and the
CHAPTER 3. METHODOLOGY

Figure 3.1: Comparison between WAVE and ETSI model according to OSI layers.[1]

DCC-management [13]

3.1 Build the simulation model

An architectural plan is needed to build an ETSI ITS simulation model that simulates the VANET environment. The ITS-S reference architecture is chosen as a guidance to build the vehicle model because it was designed by the ETSI committee. The NS3 library was chosen as a tool to build the ETSI ITS simulation model. Furthermore, the model behavior should be defined in order to achieve a better understanding about the simulation model. Finally, the simulation model needs to be validated in order to check if the simulation model behaves according to the definition.

3.1.1 Architecture Design

The ETSI ITS-Station reference architecture contains four layers, which are the Application, the Facilities, the Network & Transport, and the Access layer; and two cross layers, which are the Management and the Security layer. Figure 3.2 depicts the ITS-S reference architecture. Each layer has communication interfaces that indicate an interaction ability between layers. For example, the MS represents a communication interface between the Management and the Security layer; hence, the information on the security layer can be used by the management layer and vice versa.

The simulation model simulates the Physical, the Access, the Network & Transport, the Application, and the Management layer because these layers are described on the ETSI standard document. For the Physical layer, the simulation model simulates transceivers that operate at 5.9 GHz, OFDM and follows the IEEE 802.11p standard. The simulation model also simulates the propagation model for IVC in rural areas because a vehicle has further communication range in rural than in urban area [3]. On the Access layer, the simulation model simulates the DCC-access function that collects information on MAC and PHY layer (e.g. Channel load, Physical status, Transmit rate, etc.). For the Network & Transport layer, the simulation model simulates one-hop broadcast messages that imitate the GN SHB packets. Additionally, the one-hop broadcast
Figure 3.2: ETSI ITS-Station reference architecture [9].

message carries additional information about the sender channel load and the one-hop neighbor channel load that is described by the standard document. The detailed description about the dissemination channel load information is explained in the Channel Busy Ratio section. In the Application layer, the simulation model simulates the periodic CAM message because it is used to disseminate the channel load information to the neighbor vehicles. The cross-layer, which is the management layer, stores information that is provided by every DCC function. Therefore, an upper layer can use the information that is provided by the lower layer and vice versa [13] [10] [15].

Given the ITS-S reference architecture and the comparison ETSI model figure, the simulation model has clearer direction about which part needs to be simulated by the simulation model. Additionally, the security layer is not simulated by the model because it not stated in the standard document.

3.1.2 NS3 Simulation Modules

The current version of NS3 (i.e. ns3 3.22) does not provide a special module for the VANET ETSI model. The simulation model could have simulated the WAVE model if the wrong NS3 modules are selected. Hence, the NS3 modules that were selected to build the simulation model needs to be described in order to show that the simulation model simulates the ETSI model.

The NS3 wifi80211pHelper module is chosen to model the IEEE 802.11p network devices because this module only simulates the IEEE 802.11p both in MAC and PHY layer [26]. Meanwhile, the NS3 WaveHelper module simulates the IEEE 802.11p in MAC and PHY layer also the 1609.4 MAC extension layer [26]. Furthermore, the NS3 tracing function is chosen to simulate the DCC-access because it can provide the information about the transceiver status (e.g. PHY state, Tx power, Tx begin, Rx begin, etc.) during the simulation [5] [10]. Additionally, the DCC-access is an important component that distinguishes the ETSI ITS model from the WAVE model.

To simulate GN SHB packet with 1000 Byte payload, the simulation model should have the one-hop behavior and messages that have size 1084 Byte [14] [17] [16]. The NS3 IP broadcast
CHAPTER 3. METHODOLOGY

module was chosen to simulate the one-hop behavior. The message payload was adjusted until the message size reaches 1084 Byte. Additionally, for the channel assignment mechanism each transceiver is configured with a specific IP address (i.e. 10.1.0.0/16 for CCH, 20.1.0.0/16 for SCH1, and 30.1.0.0/16 for SCH2). Therefore, by defining a particular broadcast address as the destination address, a message will be dispatched from a particular channel. For an example, a message that has 20.1.255.255 as the destination address will be transmitted via SCH1.

The simulation model uses the NS3 UDP module to transmit the periodic CAM messages every 100ms because the CAM is broadcasted without a particular handshake procedure [13] [11] [15]. Furthermore, to simulate the management layer the simulation model uses variables that can be accessed from anywhere in the program because the management layer is a database that provides information exchange between layers [10].

3.1.3 Model Behavior

Four important factors are needed to describe the simulation model behavior. The first factor is the 802.11p configuration parameters that defines the physical hardware behavior (i.e. Transceivers). The second is the state that can occur during the simulation run time (e.g. message successfully received, message drop, etc.). The third is the initial behavior that describes the startup condition, which also determines the vehicle transmit sequence. The last factor is the type of information, which is carried by the transmitted messages because it describes the main reason why the information needs to be shared to the other vehicles.

The physical hardware behavior affects the CBR channel load measurement. Lowering the value of the carrier sense threshold will increase the message reception performance [24]. Therefore, the parameter configurations for the carrier sense threshold, noise floor, and signal noise ratio are based on a paper that also uses the CBR mechanism to determine the channel load [24]. Furthermore, in Ad-hoc network, the 802.11p uses the CSMA/CA mechanism in Distributed Coordination Function (DCF) mode to avoid message collisions. The simulation model implements the CSMA/CA broadcast message procedure algorithm because the CAM message is transmitted using GN SHB packet [2] [12]. Moreover, the Arbitrary Inter-Frame Spacing (AIFS) and the minimum and maximum Contention Window (CW) value follow the best effort access category configuration that is stated in the standard document [13]. Additionally, to avoid adjacent-channel interference between transceiver during the simulation, all vehicle transceivers are configured with the same transmit power. The transmit power is set to the maximum power that is recommended for the best effort access category because the CAM message is included as best effort access category [13]. Table 3.1 summarizes the simulation model physical hardware configuration while the flow chart of CSMA/CA broadcast message procedure algorithm can be found in the appendix B.

Second, the simulation timing model that follows the one-hop Periodic Broadcast Communication (oPBC) model from one of the reference papers [2]. Furthermore, the timing model can be divided into four categories which are the transmit time interval, the transmit and receive power, the Communication Range (CR), and the possible message conditions during the simulation. Some notations are introduced to make the simulation timing model more understandable. Assume a set $V$ of $N$ vehicles that is $v_1, v_2, v_3, \ldots, v_N$ sends periodic messages. A superscript is used to represent the $k^{th}$ occurrence of events. For an example, the $x_i^{(k)}$ represents event $x$ on vehicle $i$ occurrence $k$. 

CHAPTER 3. METHODOLOGY

Physical parameter settings

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>5 GHz</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>10 MHz</td>
</tr>
<tr>
<td>Data Rate</td>
<td>6 Mbps</td>
</tr>
<tr>
<td>Channel Number</td>
<td>180, 176, 178</td>
</tr>
<tr>
<td>Tx Power</td>
<td>23 dBm</td>
</tr>
<tr>
<td>Propagation loss</td>
<td>Three log distance</td>
</tr>
<tr>
<td>Propagation fading</td>
<td>Nakagami</td>
</tr>
<tr>
<td>SINR (SrTh)</td>
<td>8 dB</td>
</tr>
<tr>
<td>CCA Threshold</td>
<td>-95 dBm</td>
</tr>
<tr>
<td>noise Floor (nF)</td>
<td>-99 dBm</td>
</tr>
<tr>
<td>CW min, CW max</td>
<td>15, 1023</td>
</tr>
<tr>
<td>AIFS</td>
<td>110 µs</td>
</tr>
<tr>
<td>CW slot duration</td>
<td>13 µs</td>
</tr>
</tbody>
</table>

Table 3.1: The 802.11p parameter settings [3] [13] [24].

There are three time positions that can be identified when a vehicle broadcasts a periodic message. 1) The activation time that indicates the vehicle is ready to broadcast a message is represented by \( a^{(k)}_i \). 2) The start time that indicates the time when a vehicle starts transmitting a message is represented by \( s^{(k)}_i \). 3) The finish time that indicates the time when a vehicle finishes transmitting a message is represented by \( f^{(k)}_i \). Additionally, from the receiver perspective, the \( s^{(k)}_i + \delta \) and the \( f^{(k)}_i + \delta \) represent the start and finish receiving time respectively and \( \delta \) represents the air delay. The \( \delta \) can be ignored because it has a tiny value (i.e \( \delta \leq 300\,\text{ns} \)).

From the notation definition in the previous paragraph, the transmitting and receiving time interval \( t_{I}^{(k)} \) of message \( m^{(k)}_i \) can be defined as,

\[
t_{I}^{(k)} \triangleq [s^{(k)}_i, f^{(k)}_i] \quad (3.1)
\]

From the equation 3.1, we require the equation 3.2 holds for all vehicles that are transmitting periodic messages.

\[
a^{(k)}_i < s^{(k)}_i \leq f^{(k)}_i \leq a^{(k+1)}_i \quad (3.2)
\]

From the equation 3.2, the condition of no message is transmitted can be defined as \( s^{(k)}_i = f^{(k)}_i \). The no message transmit state is caused by the CSMA/CA back off mechanism that delays the message until the finish transmit time exceeds the next activation time interval. In the simulation timing model, both conditions are categorized as a message drop condition that is denoted by \( f^{(k)}_i \geq a^{(k+1)}_i \).

Whenever two vehicles are discussed (i.e. \( v_i \) and \( v_j \)), the \( i \neq j \) assumption will always hold. The \( P_{t_i}(t) \) represents the transmit power of vehicle \( v_i \), the \( P_{r_{ij}}(t) \) denotes the reception power of the vehicle \( v_j \), and the \( cP_{r_j}(t) \) describes the cumulative reception power of \( v_j \) at time \( t \). During the transmit interval \( t_{I}^{(k)} \), the \( P_{t_i}(t) > 0 \) condition is hold. According to table 3.1 the \( P_{t_i}(t) \) value is 23 dBm. Moreover, on \( v_j \), the \( cP_{r_j}(t) \) value is affected by \( P_{t_i}(t) \), the signal propagation model, and the distance between \( v_i \) and \( v_j \) at time \( t \). The \( cP_{r_j}(t) \) value is affected by the sum of all receiving power on \( v_j \) at time \( t \) plus a noise floor (i.e. \( nF \)), which denotes in the equation 3.3.
\[ cPr_j(t) = nF + \sum_{v_j} \{Pr_{ij}(t)\} \] (3.3)

In order to define the CR of \( v_i \), the definition of neighbor vehicles that can receive the message from \( v_i \) needs to be defined. The notation \( Nb_i(t) \) denotes the neighbor vehicles of \( v_i \). Therefore, the notation \( v_j \in Nb_i(t) \) describes \( v_j \) is inside the CR of \( v_i \) when \( v_i \) transmits a message on time \( t \). Furthermore, \( v_j \) is inside the CR of \( v_i \) when the ratio between \( cPr_j(t) \) and \( nF \) is equal or bigger than the single interference noise ratio (SINR). \( SrTh \) represents the SINR, and the definition of neighbor vehicles on \( v_i \) can be described as,

\[ v_j \in Nb_i(t) \triangleq \frac{Pr_{ij}(t)}{nF} \geq SrTh \] (3.4)

Additionally, the notation of neighbor vehicles is extended with an arrow down symbol to describe all neighbor vehicles of \( v_i \) during an interval \( I \).

\[ \downarrow Nb_i(I) = \bigcap_{t \in I} Nb_i(t) \] (3.5)

The last category in timing model behavior is the definition of message condition, it is an important part of timing model behavior because it describes the state of a message during the simulation. There are four message conditions,

- **Successful message received** (XMT) is the condition where a message can be received successfully by the receiving vehicle. It is a condition where the ratio value between the receipt power on \( v_j \) and the \( nF \) during the \( tI_i^{(k)} \) of message \( m_i^{(k)} \) is greater or equal to the SINR value.

\[ \forall t, t \in tI_i^{(k)} : \frac{Pr_{ij}(t)}{cPr_j(t) - Pr_{ij}(t)} \geq SrTh \] (3.6)

- **Out of Range** (OOR) is a condition where the receiving vehicles are outside the sender vehicle CR during the \( tI_i^{(k)} \) of message \( m_i^{(k)} \). It is one of the conditions when \( v_j \) does not receive the message \( m_i^{(k)} \) from \( v_i \). The OOR condition is described in equation 3.7,

\[ v_j \notin \downarrow Nb_i(tI_i^{(k)}) \] (3.7)

- **Message Drop** (MD) is the condition where the sender vehicle drops the message before it is transmitted. The idea is to prevent \( v_j \) receiving an obsolete information from \( v_i \). It is also one of the conditions when \( v_j \) does not receive message \( m_i^{(k)} \) from \( v_i \). The MD condition is described in equation 3.8,

\[ f_i^{(k)} \geq a_i^{(k+1)} \] (3.8)

- **Message Collision** (MC) is another condition where the receiving vehicles \( v_j \) does not receive message \( m_i^{(k)} \) from \( v_i \). It is a condition where the message \( m_i^{(k)} \) is corrupt because during the \( tI_i^{(k)} \) another vehicle also transmits message to \( v_j \). The condition can be determined from the receipt power of \( v_j \) during \( tI_i^{(k)} \), which is described in equation 3.9.

\[ \exists t, t \in tI_i^{(k)} : \frac{Pr_{ij}(t)}{cPr_j(t) - Pr_{ij}(t)} < SrTh \] (3.9)
Based on the message condition descriptions, the timing model of vehicles transmit and receive states during the simulation can be determined. The additional notations for the description are $T_{c_i}(k)$ that denotes the transmit condition of $v_i$, $R_{c_{ij}}(k)$ which represents the message receipt condition on $v_j$ from $v_i$, and $XMT$ that describes either the message is successfully transmitted or received, and the equations for the transmitting or receiving states are,

\[
T_{c_i}(k) = \begin{cases} 
MD & \text{if condition 3.8} \\
XMT & \text{otherwise}
\end{cases} 
\] (3.10)

\[
R_{c_{ij}}(k) = \begin{cases} 
OOR & \text{if condition 3.7} \\
MC & \text{if } v_j \in \downarrow N_b(t_i^{(k)}) \land \text{condition 3.9} \\
T_{c_i}(k) & \text{otherwise}
\end{cases} 
\] (3.11)

The third is the simulation model initial behavior that is affected by two factors (i.e. phasing and jitter). These two factors also influence the vehicle activation time during the simulation. The notation $\rho_i$ represents the phasing for $v_i$ as sender vehicle, it is drawn from a uniform random distribution that generates numbers between $0 \ldots 986\,6$ ns to randomize the activation time of $v_i$ during the initial state. This random function uses the Unix time-stamp and vehicle ID as the seed. Moreover, the notation $jitter_i(k)$ denotes another uniform random distribution that generates numbers between $0 \ldots 2 \, \mu$s to simulate the system jitter for $v_i$. The equation 3.12 describes the activation time for each vehicle during the initial state and the next activation time for $v_i$, where period denotes the value of message interval.

\[
a_i^{(k)} = \rho_i + (k \ast \text{period}) + jitter_i^{(k)}
\] (3.12)

Fourth, the type of information carried by the transmitted message. The simulation model simulates the CBR channel load information sharing based on the description in the standard document [13]. In real life, each transmitted message contains the information about the local channel load value and the sender 1-hop channel load value. These two value are identified on the receiving vehicles as receiver 1-hop channel load value and receiver 2-hop channel load value respectively [13]. However, the simulation model replaces the channel load information with the sender vehicle ID and a list of 1-hop vehicles IDs. From the receiver perspective, these two IDs are updating the list of 1-hop vehicle IDs and the list of 2-hop vehicles ID. Moreover, these vehicle IDs are used as an index to point to a two-dimensional array of channel load information. Additionally, the detailed description about the dissemination of the channel load information is explained in the Channel Busy Ratio section.

The definition of simulation model behavior in this subsection is meant to explain how the simulation model behaves. The description in this subsection shows four things: the conditions that may occur during the simulation, the 802.11p physical configuration on every vehicle, the type of information that is disseminated on the simulation model, and the initial state behavior when the simulation started.

3.1.4 Model Validation

The model validation is important because it checks the behavior of the simulation model if it deviates from the definition. The expected scenario validation is chosen to validate the model
behavior. Moreover, each module on ns3 tool had been studied and validated by the ns3 consortium. The expected scenario validation tests the simulation model with several scenarios where the results are expected. The scenarios that are chosen to validate the model are based on the definition of model behavior with some extra basic scenarios for the VANET ETSI model.

To have a better understanding about the validation scenario, the vehicles topology that is used to test the simulation model needs to be described. The topology contains three vehicles (i.e. the first, the second, and the third vehicle) that are placed in a line position. The distance between the first and the second vehicle is 250m and the distance between the first and the third vehicle is 500m while the distance between the second and the third vehicle is 250m. Therefore, for VANET CR in rural areas [3], the first and the third vehicle can only receive messages from the second vehicle, while the second vehicle can receive messages from both the first and the third vehicle. Furthermore, the validation scenarios are,

1. **The physical transceiver validation scenario**, the idea is to check if the transceivers on each vehicle are set to the right frequency, transmit power, bandwidth, and other physical parameters by checking the simulation log.

2. **Parallel transmit using different Channel scenario**, the idea is to test the simulation model if the channel segregation is working properly. The scenario sets the first and the second vehicle to transmit a message at the same time, but both vehicles use different channels from one to another. The expected result is all vehicles receive the message.

3. **The message drop scenario** tests if the vehicles in the simulation model drop the message that contains obsolete information. The scenario uses the CSMA/CA broadcast procedure that could delay a ready-to-transmit message. The scenario adjusts the length of the periodic message interval until the condition \( f_i^{(k)} \geq a_i^{(k+1)} \) is true. The scenario sets all three vehicles to use the same channel frequency and it starts from the second vehicle to transmit a message. The expected result is the first and the third have to delay their ready-to-transmit message and drop it due to obsolete information.

4. **The message out of range scenario** tests the simulation model if the OOR condition is applied. The scenario sets the distance between the first and the second vehicle greater than 300m because that is the max CR for VANET communication in rural areas [3]. The expected result from this scenario is none of the vehicles (i.e. the first and the second vehicle) receives a message.

5. **The one-hop behavior scenario** goal is to test the GN SHB packet behavior in the simulation model. The scenario allows the first vehicle to broadcast a message and the expected result is the second vehicle will not forward the message to the third vehicle.

6. **The hidden node scenario** tests the simulation model with the common wireless problem scenario. The scenario sets the first and the third vehicle using the same time and the same channel frequency to transmit messages to the second vehicle. The expected scenario is that the CSMA/CA back off algorithm is not triggered and the second vehicle does not receive any message from both vehicles.

7. **The message size validation scenario** tests if the packets, which are used in the simulation model, have the same size with the GN SHB packet with 1000 Byte payload. This

8. **The CBR information spread validation**, the idea is to check if the CBR value is spread to the neighbor vehicles. In the model behavior subsection, it is mentioned that the simulation model uses the vehicle ID as an index to access the CBR value on the two-dimensional array. Therefore, this validation scenario checks the 1-hop and the 2-hop neighbor list on one of the vehicles. Furthermore, the validation needs extra vehicles for the vehicle topology. Hence, the number of the vehicles on the line is added to the validation topology.

The purpose of validation is to check the simulation model before it is used to test the channel assignment strategies. By performing all the validation scenarios, the simulation model shows that it is ready to be used for testing the channel assignment strategies. The detail results about the validation scenario can be read in appendix A.

This section shows how the simulation model was built. It shows how the ITS-S reference architecture was implemented, and it convinces that the simulation model was represented the ETSI VANET model. This section also describes the conditions that can occur during the simulation. Finally, the simulation model also has been validated using the expected scenario method to double check the simulation model behavior.

### 3.2 Scenario Model

The scenario model is important because it defines the traffic condition, the scenario condition, and the parameters that were used for the simulation. The simulation model simulates the traffic condition of bi-directional highway during the busy hour in a rural area, where each direction has four lanes. The bi-directional highway traffic condition during busy hour is chosen because it is a common traffic condition, and the rural area is chosen so that the vehicles in the simulation model have a longer CR (i.e. 300m) [3]. By having longer CR, it provides more space to increase the number of vehicles during the simulation. Table 3.2 shows the parameters that are used for the simulation model.

<table>
<thead>
<tr>
<th>Simulation parameter settings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle velocity</td>
<td>20m/s, 15m/s, 10m/s, 5m/s</td>
</tr>
<tr>
<td>Message Size</td>
<td>1084 Byte</td>
</tr>
<tr>
<td>Message Duration</td>
<td>1.49 ms</td>
</tr>
<tr>
<td>Broadcast Period</td>
<td>10 messages per second</td>
</tr>
<tr>
<td>Max Transmit Range (CR)</td>
<td>300m (rural area)</td>
</tr>
<tr>
<td>Propagation Loss</td>
<td>Three Log distance</td>
</tr>
<tr>
<td>Propagation Fading</td>
<td>Nakagami</td>
</tr>
<tr>
<td>Phasing ($\rho$)</td>
<td>Unif [0 . . . 98(\times)6 ns]</td>
</tr>
<tr>
<td>Jitter ($jitter$)</td>
<td>Unif [0 . . . 2(\mu)s]</td>
</tr>
<tr>
<td>Simulation duration</td>
<td>10 seconds</td>
</tr>
</tbody>
</table>

Table 3.2: The simulation parameter settings [3].
CHAPTER 3. METHODOLOGY

Figure 3.3: Hidden nodes a.k.a Hidden terminal.

The scenario condition for the simulation can be divided into two types, the single domain (SD) and the multi domain (MD). These two scenario conditions represent the message collisions that are caused by the contention problem and the hidden node in the CSMA/CA broadcast procedure [2]. The contention problem appears when two or more vehicles generate the same CW value for the back off mechanism; hence, when these vehicles finish count down the CW value, they broadcast messages at the same time and it causes a collision. Furthermore, the hidden node problem occurs when a vehicle interrupts other vehicle communication because it cannot sense the existence of the transmitting vehicle. Figure 3.3 shows the hidden node collision condition.

Figure 3.4: vehicles position in simulation model scenarios

The single domain tests the channel assignment strategy in a situation where the message collision is caused only by the contention problem. In the single domain, all vehicles in the simulation model can receive the transmitted message from any vehicle. Figure 3.4 (a) shows the single domain scenario where the red dots represent the vehicles and the red dot in the middle represent an RSU. Furthermore, the multi domain tests the channel assignment strategy in a situation where the message collisions are caused by the contention problem and the hidden node problem. Figure 3.4 (b) depicts the multi domain scenario, where the red dots on the top left and bottom right are the hidden nodes.

This section shows that the proposed scenarios represent the common problems that occur in wireless communication, especially in VANET environment. Moreover, the simulation model can
only simulate in a short time because the dynamic scenario of the bi-directional highway road makes the distance of vehicles is changing; therefore, the influence of the initial condition must be minimal. It also shows the parameter settings for the simulation condition and describes why the single and multi domain scenario are chosen to test the channel assignment strategy.

3.3 Channel Busy Ratio

The channel busy ratio is a method to measure a channel load [13]. It can be interpreted as a function that calculates the ratio of incoming and outgoing messages in an interval of time (i.e. Channel Monitoring Decision Interval or CMDI). The computation about incoming and outgoing messages is provided by the DCC-access. Moreover, the CBR is triggered periodically depend on the length of the monitoring interval (CMDI).

The CBR purpose is to provide reference information for an action to reduce the number of message collisions. For example, low message rate can decrease the number of message collisions [24]. Based on the standard document, the CBR value is divided into two types. The Local CBR represents the channel load of the local vehicle. The Global CBR represents the max channel load within a 2-hop area. Additionally, the Global CBR is meant to reduce the message collisions caused by the hidden node problem [13].

3.3.1 Local Channel Busy Ratio

Every vehicle on the simulation model calculates the Local CBR; therefore, each vehicle may have different the Local CBR value. To have a clear description about the calculation of the Local CBR, some notations are introduced. The notation \( s_{mi}(k) \) represents the time when the monitoring interval starts and \( f_{mi}(k) \) denotes the time when the monitoring interval finishes in \( vi \). Furthermore, the notation \( mI_{i}(k) \) represents the length of monitoring interval in \( vi \), and it can be defined as,

\[
 mI_{i}(k) \text{ def } = [s_{mi}(k), f_{mi}(k)]
\]  

(3.13)

Moreover, the notation \( Mr_{i} \) represents the number of messages that are transmitted and received on \( vi \). Hence, referring to the model behavior definition in equation 3.10 and 3.11, the \( Mr_{i} \) in a monitoring interval \( mI \) can be described as,

\[
 Mr_{i}(mI) = |\{Tc_{i}(k)| (Tc_{i}(k) = XMT) \wedge (mI_{i}(k) \subseteq mI)\}| + |\{Rc_{ij}(k)| (Rc_{ij}(k) = XMT) \wedge (mI_{i}(k) \subseteq mI)\}|
\]  

(3.14)

Furthermore, \( Mt_{i} \) denotes the maximum number of messages that can be transmitted and received in a monitoring interval \( mI \) on \( vi \); hence, the \( Mt_{i} \) value can be described as,

\[
 Mt_{i}(mI) = \left\lfloor \frac{f_{mi}(k) - s_{mi}(k)}{f_{i}(k) - s_{i}(k)} \right\rfloor
\]  

(3.15)

The notation \( LCbr_{i} \) denotes the Local CBR value on \( vi \), and the equation 3.16 shows the definition of the Local CBR on \( vi \) in a monitoring interval \( mI \). Because there are three channels in ITS-G5A, the notations \( L_{cch}Cbr_{i} \), \( L_{sch1}Cbr_{i} \), and \( L_{sch2}Cbr_{i} \) are introduced to denote the Local CBR on CCH, SCH1, and SCH2 on \( vi \) respectively.
3.3.2 Global Channel Busy Ratio

To calculate the Global CBR value, a vehicle should receive all $LCbr_i(mI)$ value from the 1-hop and the 2-hop neighbor vehicles. The standard document defines the information sharing algorithm to disseminate the $LCbr_i(mI)$ value to the 1-hop and the 2-hop neighbor vehicles. Figure 3.5 shows the information sharing algorithm that is stated on the standard document. To have a better understanding about Figure 3.5, two assumptions need to be defined. First, the car CR can only reach the vespa, and the bike CR can only reach the vespa while the vespa CR can reach both the car and the bike. The second assumption is $LCbr_{car}(mI) < LCbr_{vespa}(mI) < LCbr_{bike}(mI)$. As a result, the $LCbr_i(mI)$ information sharing algorithm steps are,

1. The first step, each vehicle generates their own $LCbr_i(mI)$ value where in the figure 3.5 this value is labeled as CBR L0.

2. Then in the second step, every vehicle transmits their CBR L0 value and CBR L1 value, where the receiver vehicles label this value as CBR L1 and CBR L2 respectively. The CBR L1 represents the 1-hop neighbor $LCbr_i(mI)$ and the CBR L2 represents the 2-hop neighbor $LCbr_i(mI)$.

3. The third step is for the vehicles that receive more than one $LCbr_i(mI)$ value (i.e. vespa). The vespa choose the maximum $LCbr_{bike}(mI)$ value than labeled it as CBR L1.

4. Repeat the second step in the next periodic interval.

The global CBR represents the maximum value of CBR L0, CBR L1, and CBR L2 that is described in equation 3.17, where $n$ is the number of $v_i$ neighbor vehicles in the 2-hop distance and the $GCbr_i(mI)$ denotes the global CBR value in $v_i$ in a monitoring interval $mI$.

$$GCbr_i(mI) = \max(LCbr_{i}(mI), LCbr_{j}(mI), \ldots, LCbr_{n}(mI))$$

$$LCbr_i(mI) = \frac{Mr_i(mI)}{Mt_i(mI)}$$
Additionally, the CBR L0 is generated periodically based on the mi value. The CBR L1 and the CBR L2 are also generated periodically based on the message periodic interval. Therefore, the system needs synchronization phase to generate a synchronized GCbri(mi) value. For this thesis, all vehicles should have the mi value greater than 200ms. To calculate the global CBR, the optimal mi value is between 200ms - 400ms [24]. Because there are three channels in ITS-G5A, the notations GcchCbri, Gsch1Cbri, and Gsch2Cbri are introduced to denote the global CBR on CCH, SCH1, and SCH2 on v_i respectively.

3.4 Channel Assignment Strategies

The channel assignment strategy can be divided into two categories. The first category is the information base strategy that uses the vehicle physical channel information to select the proper channel (e.g. channel load, transmit power, transmit queue, etc.). The second category is the straightforward strategy, does not use the vehicle information to select the proper channel.

From the current simulation mode, three value relevant to the channel assignment mechanism: 1) The local channel load value (i.e. Local CBR) from the DCC-access [10]. 2) The max channel load value (i.e. Global CBR) in the 2-hop that is provided by the DCC-network [13]. 3) The Contention Window value that is generated by the CSMA/CA back off procedure when a channel is busy [2]. Other parameters that are provided by the DCC-access are not relevant because in the simulation model these parameters are fixed. Furthermore, the first category strategy selects the minimum value of the Local CBR, the Global CBR, and the Contention Window from each channel to decide the proper channel, which are:

**Local CBR (LCBR):**
- Select the CCH such that LcchCbri(mi) is the minimal value.
- Select the SCH1 such that Lsch1Cbri(mi) is the minimal value.
- Select the SCH2 such that Lsch2Cbri(mi) is the minimal value.

**Global CBR (GCBR):**
- Select the CCH such that GcchCbri(mi) is the minimal value.
- Select the SCH1 such that Gsch1Cbri(mi) is the minimal value.
- Select the SCH2 such that Gsch2Cbri(mi) is the minimal value.

**Least Delay (LD):**
- Select the CCH such that CWcch is the minimal value.
- Select the SCH1 such that CWsch1 is the minimal value.
- Select the SCH2 such that CWsch2 is the minimal value.

Moreover, the algorithm 1 shows the pseudo code about the channel assignment strategy when using the LCBR, the GCBR, and the LD strategy. Additionally, we assumed the local channel load value will not be disseminated to the neighbor vehicles when using the LCBR strategy; hence, there is no synchronization phase on the LCBR strategy.

The second channel assignment category is considered as another strategy that can implement the channel assignment mechanism without using any physical channel information. The Random and The Round Robin strategies are the two strategies that are able to perform channel assignments without relying on the vehicle information. Furthermore, the Random strategy defines the proper channel using the Uniform Random Distribution between the channels, and the Random...
Algorithm 1 First category channel assignment algorithm

1: procedure Transmit-LCBR or GCBR
2:   for each message $m_i$ in $v_i$ do
3:     $Cch = \text{LCBR Control Channel Or } Cch = \text{GCBR Control Channel}$
4:     $Sch1 = \text{LCBR Service Channel 1 Or } Sch1 = \text{GCBR Control Channel}$
5:     $Sch2 = \text{LCBR Service Channel 2 Or } Sch2 = \text{GCBR Control Channel}$
6:     if $Cch \leq Sch1 \& Cch \leq Sch2$ then BackOff(); Transmit message from $Cch$
7:     else if $Sch1 \leq Sch2$ then BackOff(); Transmit message from $Sch1$
8:     else BackOff(); Transmit message from $Sch2$
9:   end if
10: end for
11: end procedure

12: CwCch = 0, CwSch1 = 0, CwSch2 = 0
13: for each message $m_i$ in $v_i$ do
14:   procedure Transmit-LD(Time Triggered, CwCch, CwSch1, CwSch2)
15:     if $CwCch \leq CwSch1 \& CwCch \leq CwSch2$ then
16:       if $Cch==\text{Busy}$ then
17:         $CwCch = \text{BackOff()}$
18:         Transmit-LD (waitingTime, CwCch, CwSch1, CwSch2)
19:       else Transmit message from $Cch$
20:     end if
21:     else if $CwSch1 \leq CwSch2$ then
22:       if $Sch1==\text{Busy}$ then
23:         $CwSch1 = \text{BackOff()}$
24:         Transmit-LD (waitingTime, CwCch, CwSch1, CwSch2)
25:       else Transmit message from $Sch1$
26:     end if
27:     else
28:       if $Sch2==\text{Busy}$ then
29:         $CwSch2 = \text{BackOff()}$
30:         Transmit-LD (waitingTime, CwCch, CwSch1, CwSch2)
31:       else Transmit message from $Sch1$
32:     end if
33:   end procedure
34: end for
Robin defines the proper channel in a rotation sequence.

At the simulation initial phase, every vehicle needs to generate two variables that will determine the Round Robin channel assignment behavior. The first variable $InitCH_i$ is used to decide the first channel to transmit a message on $v_i$. The second variable $Rotation_i$ is used to decide the rotation direction of the Round Robin on $v_i$. The Random and The Round Robin behavior can be described as,

- **Random (RDM):** $unif[Cch, Sch1, Sch2]$
- **Round Robin (RBN):**
  
  $InitCH_i = unif[1, 2, 3]$, the number represents CCH, SCH1, and SCH2.
  
  $Rotation_i = unif[inc, dec]$, represents the Round Robin rotation.

$$RR_i^{(k)} = \begin{cases} 
((InitCH_i + k) \mod 3) + 1 & \text{if } Rotation_i = inc \\
((InitCH_i - k) \mod 3) + 1 & \text{if } Rotation_i = dec 
\end{cases}$$

**Algorithm 2** Second category channel assignment algorithm

1: \textbf{procedure} \textsc{Transmit}
2: \hspace{1em} for each message $m_i$ in $v_i$ do
3: \hspace{2em} CH = \text{Random() } // \text{for the random channel assignment}
4: \hspace{2em} CH = \text{RoundRobin()} // \text{for the round robin channel assignment}
5: \hspace{2em} if CH == 1 then \text{BackOff(); Transmit message from Cch}
6: \hspace{2em} else if CH == 2 then \text{BackOff(); Transmit message from Sch1}
7: \hspace{2em} else BackOff(); Transmit message from Sch2
8: \hspace{1em} end if
9: \hspace{1em} end for
10: \textbf{end procedure}

In Round Robin, the $RR_i^{(k)}$ defines the Round Robin channel assignment in $v_i$ for message sequence $k$. Moreover, to have a better understanding about the second category channel assignment, algorithm 2 shows the pseudo code about the strategy steps. The algorithm 2 is used when implementing the Random and Round Robin channel assignment strategy. Additionally, the detail code about the transmit and receive message function can be read in appendix C.

There are five channel assignment strategies that can be used in the simulation model. Three out of five strategies (i.e. Local CBR, Global CBR, and Least Delay) are based on the vehicle physical channel information that is provided by the DCC function and the random CW value from the CSMA/CA back off mechanism. The other two strategies perform the channel assignment without relying on the vehicle physical channel information.

### 3.5 Measurement Metrics

Metrics are needed to evaluate the performance shown by every channel assignment strategy. The three metrics that are chosen in this thesis are the reliability, the accuracy, and the effectiveness. The detailed description about each of these metrics is discussed in the following subsections.
3.5.1 Reliability

Every channel assignment strategy reliability can be measured by observing the successful message ratio (SMR) [2]. The successful message ratio shows how the channel assignment strategies reduce the number of message collisions. To have a better understanding of the SMR metric, other formulas are introduced. The notation \( R_{s_{ij}}(I) \) denotes the number of messages received by \( v_j \) from \( v_i \), and notation \( N_{s_{ij}}(I) \) represents the number of messages that could have been received by \( v_j \) from \( v_i \). These two notations can be described as,

\[
R_{s_{ij}}(I) = \left| \{ k | t_{I_i}^{(k)} \subseteq I \land R_{c_{ij}}^{(k)} = XMT \} \right| \quad (3.18)
\]

\[
N_{s_{ij}}(I) = \left| \{ k | t_{I_i}^{(k)} \subseteq I \land T_{c_{ij}}^{(k)} = XMT \land v_j \in v_iNb_{ij} \} \right| \quad (3.19)
\]

Therefore, the \( SMR_{ij} \) that represents the successful message from \( v_i \) to \( v_j \) in a time interval can be described as,

\[
SMR_{ij}(I) = \begin{cases} 
R_{s_{ij}}(I) & \text{if } N_{s_{ij}}(I) > 0 \\
0 & \text{if } N_{s_{ij}}(I) = 0 
\end{cases} \quad (3.20)
\]

The success message ratio on \( v_i \) is defined by notation \( SMR_i \). It can be observed by the number of messages that are received by the receiver vehicles, which is described in the equation below.

\[
SMR_i(I) = \begin{cases} 
\sum_{v_j} R_{s_{ij}}(I) & \text{if } \sum_{v_j} N_{s_{ij}}(I) > 0 \\
0 & \text{if } \sum_{v_j} N_{s_{ij}}(I) = 0 
\end{cases} \quad (3.21)
\]

Furthermore, the general success message ratio for all vehicles in the network during that particular interval can be calculated using the equation below.

\[
SMR(I) = \begin{cases} 
\sum_{v_i,v_j} R_{s_{ij}}(I) & \text{if } \sum_{v_i,v_j} N_{s_{ij}}(I) > 0 \\
0 & \text{if } \sum_{v_i,v_j} N_{s_{ij}}(I) = 0 
\end{cases} \quad (3.22)
\]

Additionally, notation \( I^{(k)} \) represents the occurrence of intervals during the simulation, which can be defined by the equation below,

\[
I^{(k)} \overset{def}{=} [k \ast period, \ (k+1) \ast period] \quad (3.23)
\]

Therefore, the average of success message ratio or \( AvSMR \) for all vehicles during the simulation can be described using the equation below, where \( i \) represents the number of intervals during the simulation and \( i,k \in \mathbb{Z}^+ \).

\[
AvSMR = \frac{1}{i} \sum_{k=1}^{i} SMR(I^{(k)}) \quad (3.24)
\]
3.5.2 Accuracy

The accuracy of channel assignment strategy can be seen from the information age on receiver vehicles. The information age measures how accurate the receiver vehicles locate the position of sender vehicle. In the ETSI model, a vehicle knows the position of the neighbor vehicles from the location table. The information in the location table is updated from every incoming message (i.e. CAM message). As an illustration, if \( v_j \) performs an action that involves the position of \( v_i \) within the duration of two incoming messages from \( v_i, v_j \) will identify the \( v_i \) position from the first message.

The notation \( k' \) denotes the nearest occurrence of a successful message receive event or \( Rc_{ij} = XMT \), after the \( Rc_{ij} = XMT \) event with a \( k \) occurrence. Moreover, the notation \( sS \) represents the time when the simulation is started and the notation \( fS \) denotes the time when the simulation is finished. Hence, the notation \( sD \) defines the simulation duration, which can be described with the equation below.

\[
sD \overset{def}{=} [sS, fS] \quad (3.25)
\]

the notation \( A_{ij} \) represents the sum of information age of \( v_i \) on the \( v_j \) location table during the simulation, which can be defined as the equation below.

\[
A_{ij}(sD) = \sum_k \{ Rc_{ij}^{(k')} - Rc_{ij}^{(k)} | (Rc_{ij}^{(k')} = XMT) \land (tI_i^{(k')} \subseteq sD) \land (tI_i^{(k)} \subseteq sD) \} \quad (3.26)
\]

Furthermore, the notation \( c_{ij}^{(k)} \) marks the condition every time \( A_{ij} \) are generated. The \( c_{ij}^{(k)} \) can be defined with the equation below,

\[
c_{ij}^{(k)} = \begin{cases} 
1 & \text{if } (Rc_{ij}^{(k')} = XMT; Rc_{ij}^{(k)} = XMT) \\
0 & \text{otherwise} 
\end{cases} \quad (3.27)
\]

Therefore, the notation \( Rm_{ij} \) that denotes the total number of \( A_{ij} \) generated by \( v_j \) from \( v_i \) during simulation can be defined with the equation below.

\[
Rm_{ij}(sD) = |\{ c_{ij}^{(k')} | (c_{ij}^{(k)} = 1) \land (tI_i^{(k')} \subseteq sD) \land (tI_i^{(k)} \subseteq sD) \}| \quad (3.28)
\]

From the definitions above, the average information age of \( v_i \) on \( v_j \) during simulation can be defined by the equation below.

\[
IA_{ij}(sD) = \frac{A_{ij}(sD)}{Rm_{ij}(sD)} \quad (3.29)
\]

The average information age of \( v_i \) in all receiving vehicles during the simulation can be described by the equation below.

\[
IA_i(sD) = \frac{\sum_{v_j} A_{ij}(sD)}{\sum_{v_j} Rm_{ij}(sD)} \quad (3.30)
\]

The average information age for all vehicles during the simulation can be described by the equation below.

\[
AvIA(sD) = \frac{\sum_{v_i,v_j} A_{ij}(sD)}{\sum_{v_i,v_j} Rm_{ij}(sD)} \quad (3.31)
\]
CHAPTER 3. METHODOLOGY

3.5.3 Effectiveness

The effectiveness is the third metric that was selected to measure the performance of channel assignment strategy. It measures the number of message drops in every vehicle during the simulation. The goal is to observe how the channel assignment strategy could reduce the number of message drops that is caused by the long delay from the CSMA/CA broadcast mechanism.

\[ Dr_i(sD) = |\{ T_{c_i}^{(k)} | T_{c_i}^{(k)} = MD \land t_{i}^{(k)} \subseteq sD \} | \]  

(3.32)

The notation \( Dr_i \) describes the number of message drops on \( v_i \) during the simulation. The notation \( AvMD_i \) denotes the average of message drops for \( v_i \) during the simulation can be calculated with the formula below, where \( n \) is the total number of messages transmitted by \( v_i \) during the simulation.

\[ AvMD_i(sD) = \frac{Dr_i(sD)}{n} \]  

(3.33)

Therefore, the average of message drops for all vehicles during the simulation can be calculated with the formula below,

\[ AvMD(sD) = \frac{\sum_{v_i,v_j} Dr_i(sD)}{\sum_{v_i,v_j} n} \]  

(3.34)

The summary of tests in this thesis

<table>
<thead>
<tr>
<th>Metrics (Row)</th>
<th>SMR</th>
<th>IA</th>
<th>MD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strategy (Column)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
<tr>
<td>RDM (2^{nd} Cat)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
<tr>
<td>RBN (2^{nd} Cat)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
<tr>
<td>LD (1^{st} Cat)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
<tr>
<td>GCBR (1^{st} Cat)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
<tr>
<td>LCBR (1^{st} Cat)</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
<td>Single and Multi</td>
</tr>
</tbody>
</table>

Table 3.3: The summary list of test.

This chapter shows that the simulation model simulates the ETSI model that is described on the standard document. This chapter also shows how the model simulates the CBR mechanism that measures the channel load and describes the two scenario models (i.e. single and multi domain). The simulation model simulates the one-Hop Periodic Broadcast communication for a VANET ETSI model, and it was described on the model behavior. The simulation model has been validated using the expected scenario method. Additionally, three metrics have been chosen to measure the performance of each channel assignment strategy in order to define the best strategy. Table 3.3 summarize all the measurement tests conducted in this thesis.
Chapter 4

Simulation Results

This chapter presents the simulation result of every strategy and analyzes their performance. The result analysis chapter is divided into five sections which are the variation of simulation result, the channel monitoring decision interval, the result of single domain scenario, the result of multi domain scenario, and the discussion. The variation of the simulation result section describes the variation range of the results after several simulations. The channel monitoring decision interval section explains the monitor interval role and how it is used to decide the channel load. Both the single and the multi domain sections present the result of each channel assignment strategy in each scenario. The discussion section discusses the result of each channel assignment strategy and suggests the proper channel assignment strategy.

4.1 The variation simulation result

Because of the random functions, the results given by the simulation model varies. Therefore, the simulation model needs to be executed more than once to observe the results pattern. In this thesis, the simulation model is executed 30 times per number of vehicles for every channel assignment strategy.

![Box plot anatomy](image)

Figure 4.1: Box plot anatomy.

Moreover, the box plot is chosen because it is the convenient way to present a group of simulation result in a graphical form. The box plot has three quadrants that represent the 25%, the
median, and the 75% of the simulation result value. The whiskers that are located above and below the box represent the maximum and minimum value of the simulation result. Therefore, the box plot depicts how accurate the result of the simulation is. The smaller the box describes the more accurate the simulation result. Additionally, the box plot represents the average value of every metric (i.e. AvSMR, AvIA, and AvMD). More detail about the box plot graphical anatomy can be seen in figure 4.1.

4.2 Channel Monitor Decision Interval

The LCBR and the GCBR strategy apply the monitoring interval as part of the channel assignment mechanism. This section describes how to determines the length of monitoring interval for both strategies. Determining the length of the monitoring interval indirectly affects the divisor value in the local CBR calculation; therefore, it affects the calculation of CBR value and influences the proper channel selection. In order to have a high percentage of the successful message ratio, the length of monitoring interval needs to be adjusted until the channel load ratio provides a value around 0.65 [7] [6] or between 0.6 to 0.7 [24]. In the simulation model, CBR is implemented as a periodic function that generates CBR value per interval (i.e. monitor interval). For example, if the length of monitoring interval is configured to 100ms, then the CBR value will be generated every 100ms. Therefore, the monitoring interval indicates how often the CBR value was generated.

![Figure 4.2: The LCBR strategy channel load between 10 to 50 ms from 30 simulation runs](image)

One of the reference papers claimed that the best monitoring interval for the GCBR strategy is between 200 - 400ms [24]. In this thesis, the length of the local channel load monitoring interval for the GCBR strategy is set to 200ms because it is the minimum value. On the other hand, the LCBR strategy does not have a synchronization phase; hence, the length of local channel load monitoring interval can be lower than 200ms. For the LCBR strategy, the length of monitoring interval is 10ms. The 10ms is chosen because it presents an optimal condition to have a high percentage of successful message ratio, which is the CBR value around 0.65 or between 0.6 to
0.7. Figure 4.2 shows the result of channel load using the LCBR strategy per channel in different length intervals.

The figure 4.2 shows using the current LCBR strategy algorithm, short monitoring interval tends to utilize the CCH more than the others while long monitoring interval tends to have an equal load for all channels. The current LCBR strategy algorithm prioritizes the CCH when defining the minimum local CBR value; hence, in a short monitoring interval, vehicles tend to use the CCH. However, in a long monitoring interval, the local CBR value is rarely updated than in the short monitoring interval. Therefore, the channel load status holds longer and caused vehicles to use a particular channel longer, which creates a pattern where all channels have an equal local CBR value (i.e. equal load for all channels).

Furthermore, the channel load shown by a long monitoring interval does not reach the lower bound of optimal channel usage (i.e. 0.6) because a long monitoring interval creates bigger divisor value; hence, it yields a small local CBR value. Additionally, each bar in figure 4.2 represents the calculation of each channel; hence, they are not affected one to another. The standard document allows vehicles to use single transceiver that only tunes on the CCH [13]; hence, prioritizing the CCH than the other channels is considered a positive feature due to the existence of single transceiver vehicles in the real world.

Further investigation was conducted to identify the possible range of optimal length of monitoring interval for the LCBR strategy. From figure 4.2, the upper bound of optimal monitoring interval can be concluded, which is the length of monitoring interval should be lower than 20ms to have high SMR result. Figure 4.3 provides the same information as in figure 4.2, but with denser granularity (i.e. 2ms). From figure 4.3, the upper and the lower bound of optimal monitoring interval can be concluded, which is the length of monitoring interval should between 6ms to 16ms to have high SMR result. Additionally, only the 10ms that managed to optimize two channels that is CCH and SCH1 above the CBR value lower bound (i.e. 0.6); hence, it is chosen as the length of monitoring interval for the LCBR strategy.

Figure 4.3: The LCBR strategy Channel load between 4 to 20 ms from 30 simulation runs

Further investigation was conducted to identify the possible range of optimal length of monitoring interval for the LCBR strategy. From figure 4.2, the upper bound of optimal monitoring interval can be concluded, which is the length of monitoring interval should be lower than 20ms to have high SMR result. Figure 4.3 provides the same information as in figure 4.2, but with denser granularity (i.e. 2ms). From figure 4.3, the upper and the lower bound of optimal monitoring interval can be concluded, which is the length of monitoring interval should between 6ms to 16ms to have high SMR result. Additionally, only the 10ms that managed to optimize two channels that is CCH and SCH1 above the CBR value lower bound (i.e. 0.6); hence, it is chosen as the length of monitoring interval for the LCBR strategy.
4.3 Single Domain Scenario

The message collision that occurs in a single domain scenario is only caused by the contention problem. Therefore, it is easier to analyze the channel assignment strategy performance using this scenario. This section presents the performance of average SMR, average MD, and average IA from all channel assignment strategies.

4.3.1 Success Message Ratio

![Graph showing SMR percentage for different channel strategies.](image)

Figure 4.4: The average SMR in single domain scenario

Figure 4.4 shows the performance of the RDM strategy. The RDM yields a good performance because the average SMR value is stable above 99%, even when the numbers of vehicles reached 150. The RDM has a good performance because it equally distributed all messages to all channels. Therefore, the multi-channel operation on the RDM strategy is similar to having a triple bandwidth.

The RBN also presented a good performance because the average SMR value is stable above 99%. Moreover, it also equally distributed all messages to all channels. Actually, the RBN strategy has two drawbacks. 1) The drawback that appears when two vehicles have a different rotation direction (i.e. increasing or decreasing). These two vehicles share the same channel every occurrence multiple of three. 2) The drawback that appears when two vehicles have the same rotation direction and starting channel. These two vehicles share the same channel on each sequence. However, in the single domain scenario these drawbacks were handled by the CSMA/CA back-off mechanism.

Unlike the two previous strategies, the LD strategy performance that is also depicted in figure 4.4 slightly decreases after the number of vehicles is above 60. The results occurred because the network started to congest when the number of vehicles is above 60. In a congested network, the back-off mechanism was triggered more often. With the current LD strategy algorithm, which tends to choose the smallest contention window value, the contention problem has a high possibility to occur.
CHAPTER 4. SIMULATION RESULTS

The next channel assignment strategy is the GCBR strategy that is also represented in figure 4.4. Similar with the LD strategy, the GCBR strategy performed well until the network was congested when the number of vehicles is above 60. In a congested network, the GCBR strategy performance decreases sharply. The decreasing performance occurred because the GCBR strategy reacted too slow to generate the global CBR value that is used to decide the proper channel. The slow reaction was due to the global CBR value is generated every 200ms.

For example, when the monitoring interval value was set to 200ms, the CBR value was generated every 200ms. Hence, if the CBR stated a condition that the CCH was not busy, then for the next 200ms a vehicle would transmit messages via CCH. The condition got worse when the number of vehicles was increased because more vehicles transmitted messages via CCH. Eventually, the CCH channel was in a busy condition, but the GCBR strategy updated this condition in the next monitoring interval (i.e. Every 200ms).

The last strategy is the LCBR that is also represented in figure 4.4. The LCBR strategy presented a good performance because the average SMR value is stable above 99%, even when the number of vehicles reaches 150. The result occurred because with 10ms monitoring intervals, the LCBR strategy reacted fast enough to generate local CBR value that avoids a vehicle to use a congested channel.

![Figure 4.5: The fairness from the top 3 strategies in single domain scenario](a) LCBR Fairness (b) RBN Fairness (c) RDM Fairness

In the single domain scenario, three strategies showed identical average SMR (i.e. LCBR, RDM, and RBN). Furthermore, the fairness of message loss on each vehicle was observed to identify which strategy performs the best among the others. Figure 4.5 represents the message loss fairness from these three strategies when the total vehicles reach 150 using the Cumulative Distribution Function. Overall, these three strategies performed a good result because the graphs form an almost ideal case. In LCBR strategy, 92% vehicles has message loss below 1%, while in RDM, 96% vehicles has message loss below 1%. The RBN strategy has slightly better fairness that the other two because 99% vehicles has message loss below 1%.

4.3.2 Message Drop

The average message drop of each strategy is depicted in figure 4.6. Except the GCBR strategy, all strategies showed a good performance (i.e. 0% of average message drop) because most strategies success to use another channel to avoid message being delay until it exceeds the next interval. The GCBR strategy has an increasing number of average message drop starts from 70 vehicles on every channel (i.e. CCH, SCH1, and SCH2). The increasing average message drop in the GCBR strategy occurs because most vehicles tend to use the same channel. Therefore, as a number
of vehicles increases, the back off gets longer, which delays the message until exceeded the next interval.

4.3.3 Information Age

The average Information Age of every strategy is presented in figure 4.7. In this metric, the strategy that provides high value has a poor performance, meaning that the information about neighbor vehicle locations is rarely updated. The IA performance is affected by the SMR and message delay because it is triggered by every message received successfully. The LCBR, RDM, and RBN strategy have a similar trend because these strategies have a similar average SMR value. These three strategies have slightly increased trends of IA as the number of vehicles increments. The increasing trends are caused by the message delay in a congested network.

The most interesting performance is presented by the LD strategy. Even though the LD strategy has a slightly lower average SMR value compared to the LCBR, RDM, and RBN; it provides the most update information for every vehicle. This is because the LD algorithm wants to dispatch messages as fast as possible and reduces the length of message delay. Meanwhile, the
number of message loss experienced by LD strategy is not large enough to affect the LD average IA performance.

A poor performance is shown by the GCBR strategy because the average IA value has significantly grown as the number of vehicles increasing. The poor performance occurred because the GCBR strategy has low average SMR value and the delay in a congested network.

4.4 Multi Domain Scenario

This section also presents the performance of average SMR, average MD, and average IA from all channel assignment strategy. The message collisions that occur in the multi domain scenario are caused by the contention and the hidden node problem. Therefore, the possibilities of the message collisions are higher in the multi domain scenario. The contention problem and the hidden node problem are common issues in wireless ad-hoc networks, especially for broadcast procedure.

4.4.1 Success Message Ratio

Figure 4.8: The average SMR in multi domain scenario

Figure 4.8 shows the average SMR from every strategy in multi domain scenario. Overall, by comparing with the SMR result from the Single Domain scenario, all strategies present a decreasing trend starts from 10 to 70 vehicles because of the hidden node problem. The effect of the hidden node problem has been started after the number of vehicles is greater than 10.

Despite the decreasing trend, the overall average SMR value that is presented by the RDM strategy are above the other strategies. The RDM performance was caused by the equal message distribution that is similar to having a triple bandwidth. Furthermore, by comparing the RDM performance from the Single Domain scenario, the decreasing pattern presented by the RDM strategy was only caused by the hidden node problem.

The performance that shows by the RBN strategy is always slightly below the RDM strategy. Besides the hidden node problem that caused the decreasing pattern, the small performance gap
between the RDM and RBN was caused by the RBN strategy drawbacks. In the multi domain scenario, the back-off mechanism could not handle the RBN drawbacks because it was not triggered by the two vehicles that are located in different domains. Additionally, the hidden node problem caused the two vehicles not able to sense the other vehicle activity before sending.

Figure 4.9: The average local CBR in the LCBR strategy from every vehicles in multi domain scenario

In general, the LCBR strategy has a less slope result in comparison to the two previous strategies. Between 10 to 40 vehicles, the LCBR strategy shows a downfall in the performance. This is possible because the LCBR tends to use the CCH before using the other channels, and it creates a high possibility of hidden node problems in the CCH to occur. However, the LCBR has better results when the number of vehicles is over 30. The LCBR strategy overtakes the RBN strategy when the vehicles are reaching 100, and ends at the same value with the RDM strategy when the vehicles are at 150 vehicles. This is because the impact of utilizing the SCH2 in LCBR strategy begins when the number of vehicles is above 30 and it leads to better results. Additionally, when the number of vehicles is below 40, the LCBR strategy uses the CCH and SCH1 with tiny usage of SCH2. Figure 4.9 supports the analysis about the LCBR strategy performance in the Multi Domain scenario.

Moreover, the performance shown by the LD strategy in the multi domain scenario is not as good as in the single domain scenario. The LD strategy performance dramatically decreases from 98% to 80% in the multi domain scenario. The result occurred because of the LD algorithm that increases the possibility of a contention problem in congested network and the hidden node problem.

The last strategy is the GCBR that has the worst performance among the other strategies. The GCBR performance sharply decreases from 98% to 64% as the number of vehicles increasing. The decreasing pattern occurred because of the hidden node problems that occurred in the multi domain scenario. Also, because of the GCBR strategy reacted too slow to generate the global CBR value that is used to avoid vehicles from using a congested channel.

The fairness between the LCBR and the RDM are compared because the LCBR was able gave
CHAPTER 4. SIMULATION RESULTS

Figure 4.10: The fairness from the LCBR and RDM strategy in Multi domain scenario

the same average SMR value when the number of vehicles is 150. Figure 4.10 depicts the Fairness comparison between the LCBR and the RDM when the number of vehicles is 150. Clearly the fairness graph shows that both strategies have the same pattern. Even though these two strategies separates the vehicles into two groups, the steepness shown by these two strategies almost identical with the ideal case. Hence, these two strategies provide a good fairness among the vehicles.

Furthermore, both strategies separated the vehicles into two groups with huge gaps of message loss: 1) The group of vehicles that in total have message loss less than 2%, which are 33% of the total vehicles in the simulation. 2) The group of vehicles that in total have message loss above 20%, which are 66% of the total vehicles in the simulation. The group that has small message loss present the vehicles that are located in the intersection domain because these vehicles can sense all vehicles when they want transmit a message. the group that has bigger message loss presents the vehicles that are located in the hidden node domains.

4.4.2 Message Drop

Figure 4.11: The average MD in multi domain scenario

The performance regarding the average message drop is presented in figure 4.11. The sub-figures depict the average message drop on each channel. Except the GCBR strategy, all strategies show zero percent of average messages drop. The result described that most strategies succeeded avoid the long message delay by using another channel. However, the GCBR strategy presented
increasing trend of the average message drop starts from 70 vehicles on every channel (i.e. CCH, SCH1, and SCH2).

The average message drop increases in the GCBR strategy because most vehicles tend to use a congested channel. The condition occurred because the GCBR strategy reacted too slow and it caused most vehicles in the same domain use the same channel. Additionally, the percentage of message drop in MD scenario is lower than the SD scenario. This is because the number vehicles per domain in MD scenario are less than the SD scenario. Therefore, the messages in MD scenario experience shorter delay compared to the SD scenario. The short delay leads to a low possibility of message drop.

### 4.4.3 Information Age

![Figure 4.12: The average IA in multi domain scenario](image)

In figure 4.12, all strategies present an increasing pattern of the average information age. The top three strategies (i.e. LCBR, RDM, and RBN) present an increasing pattern of average information age from 98ms to 140ms as the number of vehicles increasing. The increasing pattern was caused by the message delay and the decreasing average SMR value, which shows in figure 4.8, experienced by each strategy due to the hidden node problem.

The LD strategy performed poorly because it has higher average IA value compared to the top three strategies. The poor performance shown by the LD strategy was because the message loss due to the hidden node problem in the multi domain scenario. Hence, the average information age significantly increases from 100ms to 160ms in the range of 10 to 40 vehicles.

Surprisingly, the LD strategy shows a stable average information age value (i.e. 160ms) from 40 to 150 vehicles. The stable information age value occurred because the LD algorithm chooses the minimum delay (i.e. CW value). Figure 4.13 supports the analysis about the LD strategy average IA value, it depicts the average CW value that was selected in all vehicles. The average CW value shows a stable pattern after reaching 50 numbers of vehicles, which has the same pattern with the LD average IA value. Furthermore, in MD scenario, the number of vehicles in each domain
CHAPTER 4. SIMULATION RESULTS

is less than SD scenario. Hence, the network in multi domain scenario is less busy than in single scenario, and it affects the decremented time of the Contention Window.

![The average CW in all vehicles - MD scenario - run 30 times](image)

Figure 4.13: The average CW from every vehicles in multi domain scenario

The last strategy is the GCBR that has the worst performance among the others. The average information age performed by the GCBR strategy is rising as the number of vehicles increase. The increasing pattern was caused by message loss due to the hidden node and the contention problems, also the GCBR strategy that was too slow to avoid vehicles from using a congested channel.

4.5 Discussion

This section discusses the advantages and the drawbacks from each channel assignment strategy and recommends the proper channel assignment strategy for implementing GN protocol MCO on ITS-G5.

In the single domain scenario, the experiments show that the RDM and the RBN tend to equally distribute the message in all channels while the LCBR strategy tends to use the CCH because of the LCBR strategy algorithm. Additionally, in the average information age metric the LD strategy comes as the strategy that frequently update the information. The summary of the pros and cons of each strategy in single domain scenario is described in table 4.1.

In the multi domain scenario, three strategies presented some unique features: 1) The RDM presented the best results in every metric. 2) The LCBR surpassed the RBN average SMR performance and has an equal average SMR value with RDM when the number of vehicles is 150. 3) The LD presents stable information age value, even when the number of vehicles was increased. The summary of the pro and cons of each strategy in the multi domain scenario is described in table 4.2.

Both scenarios simulate the message collision conditions (i.e. the contention problem and the hidden node problem) to test the channel assignment strategy. The message collision conditions af-
### CHAPTER 4. SIMULATION RESULTS

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
</table>
| **LCBR** | • Use the CCH before using the other channel.  
• High successful message ratio. | • Information was updated less often than LD.  
• Slightly less fairness compare to RBN and RDM. |
| **RBN**  | • Slightly better fairness than to LCBR and RDM.  
• High successful message ratio. | • Not prioritize the CCH.  
• Information was updated less often than LD. |
| **RDM**  | • Slightly better fairness than to LCBR but lower than RBN.  
• High successful message ratio. | • Not prioritize the CCH.  
• Information was updated less often than LD. |
| **LD**   | • Information was updated more frequent than the other three. | • Lower successful message ratio than the other three.  
• Not prioritize the CCH. |

Table 4.1: The summary of pro and cons from each strategy in single domain scenario.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
</table>
| **RDM**  | • Presents the best performance in three metrics (i.e. SMR, MD, and IA). | • Not prioritize the CCH before use the other channel.  
• Increasing trend of average IA as the number of vehicles increased. |
| **LCBR** | • Prioritize the CCH before use the other channel. | • Increasing trend of average IA as the number of vehicles increased.  
• Perform slightly less than the RDM. |
| **LD**   | • Stable average IA value, even the number of vehicles increased to 150. | • Presents poor performance in all metrics.  
• Not optimize the CCH. |

Table 4.2: The summary of pro and cons from each strategy in multi domain scenario.
fect the channel assignment strategy performance; hence, the strategy that has better performance than the others in both scenarios is the best channel assignment strategy.

In general, the RDM strategy came as the best strategy because it showed better performance than the others in both scenarios. However, two strategies (i.e. the LC and the LCBR strategy) present a unique behavior that can be used by a specific user. The details about the specific users for both strategies are discussed in chapter 5.
Chapter 5

Conclusions and Future Works

This chapter presents the conclusion and proposes some ideas for the future works to improve the simulation model. Section 5.1 provides the summary of chapter 1, chapter 2, chapter 3, and chapter 4. In section 5.2, the important results from the simulation are presented. Finally, section 5.3 presents the list of ideas for the future works to improve the simulation model.

5.1 Summary

The Intelligent Transport System goal is to make the transportation system more robust (e.g. safe, secure, etc.). One of the methods is to allow the vehicles to communicate to each other. The ETSI (Europe) and the IEEE (US) are two organizations that create the standard model for the Inter-Vehicular Communication, which are ETSI-ITS model and WAVE model respectively. The ETSI standard document does not contain any part that explains the channel assignment mechanism to implement the MCO on the ITS-G5 frequency. The main research question is how to determine the proper channel for a packet in ITS-G5 Multi-Channel Operation. From the main question, two research questions were generated. 1) What are the possible strategies to define the proper channel? 2) What is the expected performance gain from the Multi-Channel Operation?

Chapter 2 describes the related works about the channel assignment mechanism in multi-channel operation. Two pieces of literature claimed that to increase the network capacity in MCO, the number of transceivers per vehicles should be equal to the number of channels. Other strategies that can be used to perform the channel assignment mechanism are observing the transmit queue buffer and monitoring the channel load. However, the transmit queue buffer strategy is not simulated because the simulation model assumes to have one transmit buffer. The assumption is needed to prevent an obsolete message being transmit by a vehicle.

Chapter 3 describes the methodology about how the simulation model was built. The simulation model was built based on the ETSI ITS-S reference architecture by using ns3 as a simulation tool. The simulation model simulates a highway environment in rural areas with two scenarios (i.e. single domain and multi domain). To address the first research question, all possible information that can be used for the channel assignment strategy was listed. In general, the channel assignment strategy can use the information provided by the vehicle or using the straightforward method. The RDM and the RBN are two channel assignment strategies that use the straightforward method. The LCBR, the GCBR, and the LD are three channel assignment strategies that use the vehicle...
information. In order to measure the strategies, three metrics are chosen: the average SMR, SMR, the average MD, and the average IA.

Chapter 4 discussed the simulation results from all strategies. The configuration of monitoring length for the LCBR and GCBR strategy are defined in this chapter. To address the second research question, the best channel assignment strategy needs to be defined. Furthermore, the simulation results of each strategy in the single and the multi domain scenario were analyzed in chapter 4.

5.2 Conclusion

The detailed comments on the results for each simulation are presented in chapter 4. This sub-section only describes the result based on the three measurement metrics.

- The RDM strategy
  The RDM strategy had shown a good result (i.e. stable above 99%) in the single domain scenario and came as the 1st position in the multi domain scenario (i.e. 87.6%) in the average SMR metric. The RDM strategy presented excellent results (i.e. 0%) in both single and multi domain scenarios in the average MD metric. The RDM strategy performance dropped to the 3rd position in the single domain scenario, but it came back to the 1st position in the multi domain scenario for the average IA metric.

- The RBN strategy
  The RBN had also shown a good result (i.e. stable above 99%) in the single domain scenario but came as the 3rd position in the multi domain scenario (i.e. 87%) for the average SMR metric. The RDM strategy also gave excellent performances (i.e. 0%) in both single and multi domain scenarios for the average MD metric. The RDM strategy came as the 2nd position in both single and the multi domain scenarios for the average IA metric.

- The LCBR strategy
  The LCBR presented a good result (i.e. stable above 99%) in the single domain scenario and came as the 2nd position in the multi domain scenario (i.e. 87.5%) for the average SMR metric. The LCBR strategy also showed excellent results (i.e. 0%) in both single and multi domain scenarios for the average MD metric. The LCBR strategy came as the 4th position in the single domain scenario, but came as 2nd position along with the RBN strategy in the multi domain scenario for the average IA metric.

- The LD strategy
  The LD strategy shown a fair performance (i.e. drop from 99% from 97%) in the single domain scenario and came at the 4th position in the multi domain scenario for the average SMR metric. The LD strategy had also shown impressive results (i.e. 0%) in both single and multi domain scenarios for the average MD metric. In the average IA metric, the LD strategy came as the 1st position in the single domain scenario, but it dropped to the 4th position in the multi domain scenario.

- The GCBR strategy
  From the simulation, the GCBR came as the last position in all metric measurements because the GCBR presented poor performances in all metric. In the average SMR metric, the results dropped from 99% to 75% in the single domain scenario and dropped from 99% to 64% in the multi domain scenario. In the average MD metric, the results increase from
0% to 8% in the single domain and from 0% to 4% in multi domain scenario. In the average IA metric, the results increased from 98ms to 185ms in the single domain scenario and from 98ms to 330ms in multi domain scenario.

Based on the three metrics (i.e. SMR, MD, and IA), the RDM strategy is the best strategy to perform MCO in an environment where all vehicles have multi transceivers. Furthermore, by using RDM strategy for MCO can increase the network capacity, which is the number of vehicles in a communication area. The performance that was gained from the MCO when using the RDM strategy is a high percentage of reliability, zero message drop, and an up to date information about the location of neighbor vehicles.

Meanwhile, the LCBR strategy is better for users that accept the consequence of having less performance than the RDM strategy in order to add other type of messages (e.g. management message, etc) beside the CAM message. Prioritizing the CCH made the other channels are less busy than the CCH; thus, these less busy channels can be used for broadcasting the other type of messages. Additionally, the LCBR strategy also provides high network capacity and zero message drop, but less reliability and accuracy compared to the RDM strategy.

The LD strategy is best for users that only concern about predictability and ignore the strategy performance. It suits for users that want to use the neighbor vehicles position information for real-time applications. The simulation result shows the LD strategy has predictable IA value (i.e. 160ms) when the number of vehicles is above 30. Meanwhile, when the number of vehicles is less than 30, a compensating delay can be added to have a predictable IA value.

5.3 Future Works

There are several things that can be done in the future to improve the simulation model. Some of these improvements are based on the findings during the simulation.

1. **The vehicles movement.** The current model can only be executed in a short amount of time. The bi-directional traffic causes the vehicles to move out of the scene if the simulation was executed in a long duration. To improve the limitation of the condition, the vehicle movement can be changed. A large circular movement for every vehicle can remove the short duration limitation in the simulation model.

2. **Implement filtering for the channel load calculation.** The current simulation model implements the calculation of channel load without filtering. In the real world, filtering is a common technique that is used in wireless communication to take out random fluctuations. One of the examples is by using the moving average that makes the channel load value gets more influence from newer events. Another purpose of filtering is to smoothen the fluctuation of channel load value (e.g. channel load = old channel load * 0.5 + new channel load * 0.5).

3. **Extra domain in the multi domain scenario.** In the current simulation model, there are three domains in the multi domain scenario. The multi domain scenario contains the hidden node problem; however, the vehicles that are located in the intersection domain still can sense the status of all vehicles. It can be seen from the fairness CDF result of the LCBR strategy and the RDM strategy. To remove this limitation, each vehicle should have their own hidden node vehicle; hence, an extra domain needs to be added as the hidden node for the intersection vehicles.
4. **Single transceiver vehicle.** The simulation results show that the RDM strategy is the best strategy to perform MCO in an environment where all vehicles have multi transceivers. The standard documents state that vehicles may use a single transceiver when implementing the GN protocol in ITS-G5. The single transceiver vehicles should only tune their transceiver to the CCH. It would be interesting to test the RDM strategy performance in an environment that contains several single transceiver vehicles. For example, a single domain scenario where half of the total vehicles use a single transceiver.
Bibliography


Appendix A

Model Validation

1. The Physical transceiver validation scenario

- The first line in each the figure represents the modulation, the Data Rate, the Bandwidth.
- The second line in each the figure represents the channel frequency, the channel number.
- The third line in each the figure represents the transmit power (txPower).

Figure A.1: SCH1

Figure A.2: SCH2

Figure A.3: CCH
APPENDIX A. MODEL VALIDATION

2. Parallel transmit using different Channel scenario

![Validation topology](image)

- The green arrow represents message sending.

![Parallel transmit scenario](image)

- The first 3 lines show the time, the event, the vehicle that conduct the event, and message number.
- The last 4 lines show the time, the event, the sender vehicle ID, the receiver vehicle ID, and message number.

3. The message drop scenario

![Message drop scenario](image)

- The period parameter in this scenario is configured to 0.5 ms.
- The first line shows vehicles ID 0 send message to neighbor vehicles.
- The second and the third lines show vehicle ID 1 and 2 delay the message until it exceeds the next activation time then drop the message.
- The fourth and fifth lines show the vehicle ID 1 and 2 finished received the message that had been transmitted by vehicle ID 0 in the first line.
4. The message out of range scenario

![Figure A.7: Message out of range scenario.](image)

- The three lines show all vehicles send a message but none of the vehicles received the message.

![Figure A.8: Message out of range scenario in detail.](image)

- The white underlines show the minimum and the maximum distance between vehicles.
- The red underlines show the reason of the message does not delivered.
5. The one-hop behavior scenario

Figure A.9: one-hop behavior scenario topology

- The figure shows vehicle ID 1 success to send message to vehicle ID 0.
- Vehicle ID 0 does not forward the message to vehicle ID 2.

Figure A.10: one-hop behavior scenario

- The first line presents the vehicle ID 1 send a message
- The second line presents the vehicle ID 0 receive the message

6. The hidden node scenario

Figure A.11: Hidden node scenario

- The first and second lines show that vehicle ID 1 and 2 send message at almost the same time and show that the vehicle ID 0 did not receive the message.
7. The message size validation scenario

- The second line shows the packet size that is 1084 Byte.

8. The CBR information spread validation

- Figure A.14 shows the generic location table of each vehicles.
- X:Y, where X represents the local vehicle ID and Y represent the neighbor vehicle ID.
- When Y = -1, it represents empty slot.
Appendix B

CSMA/CA Broadcast Procedure

Figure B.1: CSMA/CA Broadcast mode procedure [2].
Description about the CSMA/CA figure.

- The 'a' holds the AIFS value.
- The 'c' holds the Contention Window value.
- The 'do,c' is boolean value that indicate whether the back off need to be performed.
- The 'CH' defines the channel status.
Appendix C

Transmit and Receive Message Source Code

C.1 Transmit message source code

```cpp
static void GenerateTraffic (Ptr<Socket> socket, uint32_t pktSize, uint32_t MaxPkt,
        Time pktInterval, uint32_t pktCount)
{
    std::stringstream msgx; // packet sequence information;
    int iCarID = socket->GetNode()->GetId();

    if (pktCount <= MaxPkt)
    {
        Time packetInterval = MilliSeconds(interval);

        // executed once per packet
        if (tempPktCount[iCarID] != pktCount)
        {
            sendPayload[iCarID] = PayloadToString(iCarID, numVehicles);
            packeActivationTime[iCarID] = Simulator::Now();
            tempPktCount[iCarID] = pktCount; // guard
            if (dGlobalCBR[iCarID][0] < dGlobalCBR[iCarID][1] && dGlobalCBR[iCarID][0] <
                dGlobalCBR[iCarID][2]) { channelAssign[iCarID][pktCount] = 1; }
            else if (dGlobalCBR[iCarID][1] < dGlobalCBR[iCarID][2]) { channelAssign[iCarID][
                pktCount] = 2; }
            else { channelAssign[iCarID][pktCount] = 3; }
            packetJitter[pktCount] = MicroSeconds(jitter(iCarID, pktCount));
            Simulator::Schedule (packetJitter[pktCount], &GenerateTraffic, socket,
                    pktSize, MaxPkt, pktInterval, pktCount);
        }
        else
        {
            // prepare to send
            msgx << sendPayload[iCarID]; // Send payload Location Table extension to
            receiver.

            // Round Robin
            uint32_t RR;
            if (RobinIncrease[iCarID] == 1) { RR = (RoundRobin[iCarID] - 1) + pktCount; }
            else { RR = (RoundRobin[iCarID] - 1) + ((MaxPkt + 1) - pktCount); }
        }
    }
```
if /*RDM, RBN, LCBR, GCBR, LD => CCH*/{
    if ((packeActivationTime[iCarID]+packtInterval)−Simulator::Now() < 
        MsgDuration){
        DropPktCounter[iCarID][0]++; 
        std::cout << Simulator::Now() << ": Drop event, on VehicleID " << socket->
        GetNode()−>GetId() << ", Msg number:" <<pktCount 
        "\t total msg drop:" << DropPktCounter[iCarID][0] << std::endl; 
        Simulator::Schedule (pktInterval−packetJitter[pktCount],
        &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount+1);
    } else if (aifs[iCarID][pktCount][0] > 0){
        aifs[iCarID][pktCount][0]−−;
        if (isPhyBusy[iCarID][0] == true){
            doBf[iCarID][pktCount][0] = true;
            aifs[iCarID][pktCount][0] = Aifs;
            Simulator::Schedule (aifsTimeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else{ 
            Simulator::Schedule (aifsTimeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        }
    } else{ 
        if (doBf[iCarID][pktCount][0] == true &
            & bf[iCarID][pktCount][0] == −1){
            GenerateBF (iCarID, pktCount, 0);
        }
        if (isPhyBusy[iCarID][0] == true) {
            aifs[iCarID][pktCount][0] = Aifs;
            Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else if (bf[iCarID][pktCount][0] > 0){
            bf[iCarID][pktCount][0]−−;
            Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else{ 
            socket−>SetAllowBroadcast (true);
            socket−>Connect (InetSocketAddress (Ipv4Address ("10.1.255.255"), port));
            SenderPosition[iCarID] = socket−>GetNode()−>GetObject<
                MobilityModel>()−>GetPosition(); Record position when sending the packet
            socket−>Send (Create<
                Packet> ((uint8_t*)msgx.str().c_str(),c,str(),pktSize));
            std::cout << Simulator::Now() << "\t Sending event, By VehicleID:" << 
                socket−>GetNode()−>GetId() << ", Msg number:" << pktCount << std::endl;
            Simulator::Schedule (pktInterval−packetJitter[pktCount],
                &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount+1);
        }
    }
}
else if /*RDM, RBN, LCBR, GCBR, LD => SCH*/{
    if ((packeActivationTime[iCarID]+packtInterval)−Simulator::Now() < 
        MsgDuration){
        DropPktCounter[iCarID][1]++; 
        std::cout << Simulator::Now() << ": Drop event, on VehicleID " << socket−>
        GetNode()−>GetId() << ", Msg number:" << pktCount 
        "\t total msg drop:" << DropPktCounter[iCarID][1] << std::endl; 
        Simulator::Schedule (pktInterval−packetJitter[pktCount],
        &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount+1);
    } else if (Aifs[iCarID][pktCount][0] > 0){
        Aifs[iCarID][pktCount][0]−−;
        if (isPhyBusy[iCarID][0] == true){
            doBf[iCarID][pktCount][0] = true;
            Aifs[iCarID][pktCount][0] = Aifs;
            Simulator::Schedule (aifsTimeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else{ 
            Simulator::Schedule (aifsTimeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        }
    } else{ 
        if (doBf[iCarID][pktCount][0] == true &
            & bf[iCarID][pktCount][0] == −1){
            GenerateBF (iCarID, pktCount, 0);
        }
        if (isPhyBusy[iCarID][0] == true) {
            Aifs[iCarID][pktCount][0] = Aifs;
            Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else if (bf[iCarID][pktCount][0] > 0){
            bf[iCarID][pktCount][0]−−;
            Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, 
                MaxPkt, pktInterval, pktCount);
        } else{ 
            socket−>SetAllowBroadcast (true);
            socket−>Connect (InetSocketAddress (Ipv4Address ("10.1.255.255"), port));
            SenderPosition[iCarID] = socket−>GetNode()−>GetObject<
                MobilityModel>()−>GetPosition(); Record position when sending the packet
            socket−>Send (Create<
                Packet> ((uint8_t*)msgx.str().c_str(),c,str(),pktSize));
            std::cout << Simulator::Now() << "\t Sending event, By VehicleID:" << 
                socket−>GetNode()−>GetId() << ", Msg number:" << pktCount << std::endl;
            Simulator::Schedule (pktInterval−packetJitter[pktCount],
                &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount+1);
        }
    }
}
APPENDIX C. TRANSMIT AND RECEIVE MESSAGE SOURCE CODE

```cpp
} else if (aifs[iCarID][pktCount][1] > 0) {
    aifs[iCarID][pktCount][1] = -1;
    if (isPhyBusy[iCarID][1] == true) {
        do_bf[iCarID][pktCount][1] = true;
        aifs[iCarID][pktCount][1] = Aifs;
        Simulator::Schedule(aifsTimeSlot, &GenerateTraffic, socket, pktSize,
                             MaxPkt, pktInterval, pktCount);
    } else {
        Simulator::Schedule(aifsTimeSlot, &GenerateTraffic, socket, pktSize,
                             MaxPkt, pktInterval, pktCount);
    }
} else {
    if (do_bf[iCarID][pktCount][1] == true && bf[iCarID][pktCount][1] == -1) {
        GenerateBF(iCarID, pktCount, 1);
    }
    if (isPhyBusy[iCarID][1] == true) {
        aifs[iCarID][pktCount][1] = Aifs;
        Simulator::Schedule(timeSlot, &GenerateTraffic, socket, pktSize,
                             MaxPkt, pktInterval, pktCount);
    } else if (bf[iCarID][pktCount][1] > 0) {
        bf[iCarID][pktCount][1] = -1;
        Simulator::Schedule(timeSlot, &GenerateTraffic, socket, pktSize,
                             MaxPkt, pktInterval, pktCount);
    } else {
        socket->SetAllowBroadcast(true);
        socket->Connect(InetSocketAddress(Ipv4Address("20.1.255.255"), port));
        senderPosition[iCarID] = socket->GetNode()->GetPosition(); // Record position when sending the packet
        socket->Send(CreatePacket((uint8_t*)msgx.str().c_str(), pktSize));
        // std::cout << Simulator::Now() << " : Sending event, By VehicleID: " << socket->GetNode()->GetId() << " , Msg number: " << pktCount << std::endl;
        Simulator::Schedule(pktInterval - packetJitter[pktCount], &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount + 1);
    }
}
} else /-RDM, RBN, LCBR, GCBR, LD => SCH2/- {
    if ((packeActivationTime[iCarID]+packeInterval) - Simulator::Now() <
         MsgDuration) {
        DropPktCounter[iCarID][2]++;
        std::cout << Simulator::Now() << " : Drop event, On VehicleID " << socket->GetNode()->GetId() << " , Msg number: " << pktCount
                 << " , total msg drop: " << DropPktCounter[iCarID][2] << std::endl;
        Simulator::Schedule(pktInterval - packetJitter[pktCount], &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount + 1);
    } else if (aifs[iCarID][pktCount][2] > 0) {
        aifs[iCarID][pktCount][2] = -1;
        if (isPhyBusy[iCarID][2] == true) {
            do_bf[iCarID][pktCount][2] = true;
            aifs[iCarID][pktCount][2] = Aifs;
            Simulator::Schedule(aifsTimeSlot, &GenerateTraffic, socket, pktSize,
                                 MaxPkt, pktInterval, pktCount);
        } else {
            // further code
        }
    } else {
        // further code
    }
}
```

60 Design and Evaluation of Multi-Channel Operation Implementation of ETSI GeoNetworking Protocol for ITS-G5
APPENDIX C. TRANSMIT AND RECEIVE MESSAGE SOURCE CODE

Simulator::Schedule (aifsTimeSlot, &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount);
}
else {
    if (do_bf[iCarID][pktCount][2] == true & & bf[iCarID][pktCount][2] == 1) {
        GenerateBF (iCarID, pktCount, 2);
    if (isPhyBusy[iCarID][2] == true) {
        aifs [iCarID][pktCount][2] = Aifs;
        Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount);
    } else if (bf[iCarID][pktCount][2] > 0) {
        bf[iCarID][pktCount][2]--;
        Simulator::Schedule (timeSlot, &GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount);
    } else {
        socket->SetAllowBroadcast (true);
        socket->Connect (InetSocketAddress (Ipv4Address ("30.1.255.255"), port));
        SenderPosition[iCarID] = socket->GetNode()->GetObject<MobilityModel>() ->GetPosition(); Record position when sending the packet
        socket->Send (CreatePacket>((uint8_t*)msg_x.str().c_str(), pktSize));
        std::cout << Simulator::Now() << ": Sending event, By VehicleID:" << socket->GetNode()->GetId() << ", Msg number:" << pktCount << std::endl;
        Simulator::Schedule (pktInterval-packetJitter[pktCount], & GenerateTraffic, socket, pktSize, MaxPkt, pktInterval, pktCount+1);
    }
}
for (int i = 0; i < numVehicles; i++) {Age[i][iCarID] += (Simulator::Now() - MilliSeconds(98.5)) - LastInfoUpdate[i][iCarID];
    socket->Close();
}
C.2 Receive message source code

```cpp
void ReceivePacket ( Ptr<Socket> socket )
{
    Ptr<Packet> packet;
    Address from;
    uint32_t iReceiverID = socket->GetNode() -> GetId();
    Time pktInterval = MilliSeconds (interval);

    while ( packet = socket->RecvFrom (from) )
        {
            // = = = = Extract Payload (i.e. TimeStamp and sender LocT) from the incoming packet and merge with the LocTable = = = = = = = =
            uint8_t *buffer = new uint8_t [packet->GetSize()];
            packet->CopyData (buffer, packet->GetSize());
            std::string Payload = std::string((char*)buffer);
            std::string timeinfo = ExtractTimeStamp (Payload);
            boost::erase_head(Payload, timeinfo.length() +1);
            boost::replace_first(timeinfo, "ns", "ns");
            boost::replace_all(timeinfo, " ");
            double dummy = boost::lexical_cast<double>(timeinfo);
            Time pktTimeStamp = NanoSeconds(dummy);

            // std::cout << pktTimeStamp << " Time stamp : Information age : " <<\n            // (Simulator::Now() - pktTimeStamp) << std::endl;

            boost::replace_all(Payload, ",", " ");

            int i = 0;
            std::stringstream ssin(Payload);
            while ( ssin.good() && i < numVehicles+1) {
                senderID
                ssin >> tempExtLocTable [iReceiverID][i];
                i++;
            }

            // = = = = Merge incoming message 0 hop info and covert as 1 hop info = = = = = = = = =
            int ext1hopCounter = 0;
            do{
                if (tempExtLocTable[iReceiverID][0]==ExtLocTable1hop[iReceiverID][ext1hopCounter])
                    {break;}
                else if (ExtLocTable1hop[iReceiverID][ext1hopCounter]==-1)
                    {ExtLocTable1hop[iReceiverID][ext1hopCounter]=tempExtLocTable[iReceiverID][0];break;}
                ext1hopCounter++;
                while(true);

            // = = = = Merge incoming message 1 hop info and covert as 2 hop info = = =
            int tempCounter=1;
```
APPENDIX C. TRANSMIT AND RECEIVE MESSAGE SOURCE CODE

```
int extCounter = 0;

while (tempExtLocTable[iReceiverID][tempCounter] != -1) {
    ExtLocTable2hop[iReceiverID][extCounter] = tempExtLocTable[iReceiverID][tempCounter];
    if (ExtLocTable2hop[iReceiverID][extCounter] == tempExtLocTable[iReceiverID][tempCounter] && ExtLocTable2hop[iReceiverID][extCounter] == -1) {
        break;
    } else if (ExtLocTable2hop[iReceiverID][extCounter] == tempExtLocTable[iReceiverID][tempCounter]) {
        break;
    } else {
        extCounter++;
    }
}

Ipv4Address sourceIPadd = InetSocketAddress::ConvertFrom(from).GetIpv4();
VectorReceiverPosition = socket->GetNode()->GetObject<MobilityModel>()->GetPosition();

// Extract sender ID from the packet, = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
std::string trim = boost::lexical_cast<std::string>(sourceIPadd);
std::string trimCount = boost::lexical_cast<std::string>(sourceIPadd);
boost::erase_head(trim, 7);
int tempID = boost::lexical_cast<int>(trim);
if (tempID < 10) {
    boost::erase_head(trimCount, 5);
    boost::erase_tail(trimCount, 2);
} else if (tempID >= 10 && tempID < 100) {
    boost::erase_head(trimCount, 5);
    boost::erase_tail(trimCount, 3);
} else {
    boost::erase_head(trimCount, 5);
    boost::erase_tail(trimCount, 4);
}
int CountID = boost::lexical_cast<int>(trimCount);
int SenderID = (CountID * 256) + tempID - 1;

// calculate distance between sender and receiver = = = = = = = = = = = = = = = = = = = = = = = = =
// float distance = sqrtf((std::pow((SenderPosition[SenderID].x - ReceiverPosition.x).2)) + (std::pow((SenderPosition[SenderID].y - ReceiverPosition.y).2)));
ReceivePktCounter[iReceiverID][SenderID]++;

if (ReceivePktCounter[iReceiverID][SenderID] < numOfPackets) {
    if (LastInfoUpdate[iReceiverID][SenderID] != 0) {
        Age[iReceiverID][SenderID] += (Simulator::Now() - LastInfoUpdate[iReceiverID][SenderID]);
        LastInfoUpdate[iReceiverID][SenderID] = Simulator::Now();
    } else {
        LastInfoUpdate[iReceiverID][SenderID] = Simulator::Now();
    }
}
```
APPENDIX C. TRANSMIT AND RECEIVE MESSAGE SOURCE CODE

```cpp
std::cout << Simulator::Now() << " : Receive event, from :" 
<< SenderID << ", to : " 
<< iReceiverID //<< " , Distance sender & receiver:" 
// << distance 
<< ", Total Messages : " << ReceivePktCounter[iReceiverID][SenderID] << std::endl;
```

Listing C.2: Receive Message Code