

University of Windsor

## Scholarship at UWindor

---

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

---

2018

### Adaptive Power Level for DSRC Congestion Control

Maan Joseph

*University of Windsor*

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

---

#### Recommended Citation

Joseph, Maan, "Adaptive Power Level for DSRC Congestion Control" (2018). *Electronic Theses and Dissertations*. 7366.

<https://scholar.uwindsor.ca/etd/7366>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email ([scholarship@uwindsor.ca](mailto:scholarship@uwindsor.ca)) or by telephone at 519-253-3000ext. 3208.

**Adaptive Power Level for DSRC Congestion Control**

by

**Maan Joseph**

**A Thesis**

**Submitted to the Faculty of Graduate Studies  
through the School of Computer Science  
in Partial Fulfillment of the Requirements for  
the Degree of Master of Science  
at the University of Windsor**

**Windsor, Ontario, Canada**

**2018**

**© 2018 Maan Joseph**

**Adaptive Power Level for DSRC Congestion Control**

by

**Maan Joseph**

**APPROVED BY:**

**A. Sarker**

**Department of Mathematics and Statistics**

---

**S. Mavromoustakos**

**School of Computer Science**

---

**R. Kent, Co-Advisor**

**School of Computer Science**

---

**A. Jaekel, Advisor**

**School of Computer Science**

---

**February 12, 2018**

# Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

# Abstract

Vehicular industries and researchers have invested efforts to reduce avoidable accidents through the means of Vehicle to Vehicle (V2V) wireless communication using Vehicular Ad Hoc Networks (VANETs) through the periodic exchange of Basic Safety Messages (BSMs). The transmission rate of BSMs is defined by IEEE 1609 to be 10 Hz. With a high vehicular density, Network Congestion can quickly arise in the 5.9 GHz spectrum, rendering the system as unreliable because safety messages are not delivered on time. Researchers have focused on altering the rate of transmission and/or power of transmission in congestion control algorithms. The rate of transmission dictates how many messages each vehicle sends per second. Further, the transmission power dictates how far each message travels; it is known that messages transmitted with higher power will reach further distances. Based on that, our algorithm performs two operations to mitigate channel congestion; a) we send a number of low powered packets based on the node's velocity, the higher the velocity then the higher transmission power, then followed by a high powered packet to maintain awareness for distant vehicles, b) we increase the power of transmission in a cyclic fashion. By doing so, we can maintain necessary level of awareness for closer vehicles, while sacrificing some awareness for distant ones. The goal is to provide adequate awareness for all vehicles, while reducing the overall congestion of the wireless channel.

# Dedication

I dedicate this thesis to my parents Dr. Mumtaz Anayee and Dr. Ghada Misho, without whom I would not have made it this far. To my fiancée Linda Pio, who always pushed me and believed in me, and for the support of my siblings Mada, May and Majid.

# Acknowledgements

First I would like to thank God for giving me the strength and ability to pursue my undergraduate and graduate degrees.

I would like to express my sincere gratitude to my supervisors Dr. Arunita Jaekel and Dr. Robert Kent, for their motivation, enthusiasm and immense knowledge. Thank you for your invaluable time and support during my undergraduate and graduate degrees. The knowledge and skills you have given me will last me a lifetime, for that, I cannot thank you enough.

I would also like to thank the rest of my committee members, Dr. Animesh Sarker and Dr. Stephanos Mavromoustakos for their help and support.

I would like to thank my family for sticking with me, Muntaz, Ghada, Mada, May, Majid, my fiancée Linda and the rest of my extended family and friends, I love you all dearly.

For the Computer Science department faculty and staff, especially secretaries on the 5th floor of Lambton tower, University of Windsor, Karen, Gloria, Tina and Melissa, thank you.

For the people who are not with us today, my grandparents especially my grandfather Dr. Fawzi Misho and Linda's father Mr. Basel Pio, please continue to watch over us from heaven.

*Take delight in the Lord, and he will give you the desires of your heart.*

Psalm 37:4

# Contents

<b>Declaration of Originality</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Dedication</b>	<b>v</b>
<b>Acknowledgements</b>	<b>vi</b>
<b>List of Tables</b>	<b>x</b>
<b>List of Figures</b>	<b>xi</b>
<b>Abbreviations</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Motivation . . . . .	2
1.3 Problem Statement . . . . .	4
1.4 Solution Outline . . . . .	5
1.5 Thesis Organization . . . . .	6
<b>2 Background</b>	<b>7</b>
2.1 Intelligent Transportation System . . . . .	7
2.2 Vehicle-to-Vehicle and Vehicle-to-Infrastructure . . . . .	9
2.2.1 Vehicle to Vehicle (V2V) . . . . .	10



		viii
2.2.2	Vehicle to Infrastructure (V2I) . . . . .	10
2.2.3	Hybrid Scenario V2X . . . . .	11
2.3	Current Problems and Solutions . . . . .	13
2.3.1	High Node Density in MAC OSI Stack . . . . .	13
2.3.2	Choosing congestion control parameters . . . . .	13
2.3.3	Simulation . . . . .	14
2.4	DSRC Challenges of interest to our Research . . . . .	14
2.5	Terminology . . . . .	16
2.6	Algorithm Performance Analysis . . . . .	18
2.7	Literature Review . . . . .	19
<b>3</b>	<b>Proposed Power Adaptation algorithm</b>	<b>23</b>
3.1	Introduction . . . . .	23
3.2	Synopsis of Network Congestion Control . . . . .	24
3.3	Architecture of proposed algorithm . . . . .	25
3.3.1	SpeedFactor . . . . .	25
3.3.2	Expansion of Awareness . . . . .	26
3.4	High Level Outline . . . . .	27
<b>4</b>	<b>Experiment and Results</b>	<b>32</b>
4.1	Simulation . . . . .	32
4.1.1	Simulation Setup . . . . .	33
4.1.2	Simulation Runs . . . . .	34
4.2	Results . . . . .	36
4.2.1	Packets Sent . . . . .	37
4.2.2	Packets Received . . . . .	38
4.2.3	Lost Packets . . . . .	39
4.2.4	Beacon Receive Rate (BRR) . . . . .	40

4.2.5	Beacon Error Rate (BER) . . . . .	ix 41
4.2.6	Channel Busy Time . . . . .	42
4.2.7	Inter-Packet Delay (IPD) . . . . .	43
4.2.8	Fairness . . . . .	45
<b>5</b>	<b>Conclusion and Future Work</b>	<b>47</b>
5.1	Conclusion . . . . .	47
5.2	Future Work . . . . .	48
<b>A</b>	<b>OMNeT Configuration</b>	<b>49</b>
<b>B</b>	<b>SUMO Configuration</b>	<b>52</b>
<b>C</b>	<b>Algorithm Source File</b>	<b>53</b>
<b>D</b>	<b>CSMA Flowchart</b>	<b>54</b>
<b>E</b>	<b>Scalar Results</b>	<b>55</b>
	<b>Bibliography</b>	<b>57</b>
	<b>Vita Auctoris</b>	<b>61</b>

# List of Tables

Table 2.1 Supported applications of VANET . . . . .	12
Table 2.2 List of simulators used by researchers . . . . .	14
Table 2.3 BSM Part One Structure . . . . .	17
Table 2.4 Classifications of Congestion Control Algorithms . . . . .	22
Table 3.1 Transmission Powers chosen for each vehicle . . . . .	29
Table 4.1 Simulation Parameters . . . . .	33
Table 4.2 Speed Factor Assignment . . . . .	36
Table 4.3 Number of vehicles observed . . . . .	37
Table E.1 10 Hz Scalar Results . . . . .	55
Table E.2 OSC Scalar Results . . . . .	56
Table E.3 Adaptive Scalar Results . . . . .	56
Table E.4 Fairness . . . . .	56

# List of Figures

Figure 1.1 Layered architecture for DSRC communications . . . . .	3
Figure 2.1 ITS Applications . . . . .	8
Figure 2.2 List of Frequencies dedicated for the type of message transmitted . . .	9
Figure 2.3 Hybrid Ad-Hoc Network:Two separated zones exchange information using an RSU. [22] . . . . .	12
Figure 2.4 DSRC Structure . . . . .	15
Figure 3.1 Vehicles with different speeds . . . . .	26
Figure 3.2 Increasing power of transmission for consecutive packets . . . . .	26
Figure 3.3 Adaptive Algorithm . . . . .	27
Figure 3.4 Desired Travel distance of transmitted packets . . . . .	29
Figure 3.5 Initialize method called at vehicle creation . . . . .	30
Figure 3.6 Adaptive method called before every packet is transmitted. Procedure 2.	30
Figure 3.7 Adaptive method called before every packet is transmitted. Procedure 1.	31
Figure 4.1 Mobile OMNeT++ Nodes Representation . . . . .	34
Figure 4.2 Scenario: Detroit, Michigan, U.S. . . . .	35
Figure 4.3 Scenario: Windsor, Ontario, Canada . . . . .	35
Figure 4.4 Total Sent Packets . . . . .	37
Figure 4.5 Total Received Packets . . . . .	38
Figure 4.6 Total Lost Packets . . . . .	39
Figure 4.7 Beacon Receive Rate . . . . .	40

Figure 4.8 Beacon Error Rate . . . . .	xii
Figure 4.9 Channel Busy Ratio (CBR) . . . . .	41
Figure 4.10 Six Lane Inter-Packet Delay . . . . .	42
Figure 4.11 12 Lane Inter-Packet Delay . . . . .	44
Figure 4.12 E.C. Row Inter-Packet Delay . . . . .	44
Figure 4.13 Interstate i-75 Inter-Packet Delay . . . . .	45
Figure 4.14 Algorithm Fairness . . . . .	45
Figure 4.15 Jain's Fairness Index . . . . .	46
Figure B.1 SUMO Route Configuration . . . . .	46
Figure B.2 SUMOCFG . . . . .	52
	52

# Abbreviations

BER	Beacon Error Rate
BRR	Beacon Reception Rate
BSM	Basic Safety Message
CBR	Channel Busy Ratio
CTS	Clear-to-Send
DCC	Decentralized Congestion Control
DSRC	Dedicated Short Range Communication
FCC	Federal Communications Commission
IPD	Inter-Packet Delay
ITS	Intelligent Transportation System
MAC	Medium Access Control
NHTSA	National Highway Traffic Safety Administration
OBU	On Board Unit
OSC	Oscillating Power Control
OSM	Open Street Map
RSU	Road Side Unit
RTS	Request-to-Send
TE	Tracking Error
TTC	Time to Collision
US-DOT	United States Department of Transportation
VANET	Vehicular Ad-Hoc Network
VEINS	Vehicles In Network Simulation
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
V2X	Vehicle to Anything
WAVE	Wireless Access in Vehicular Environment
WSM	Wave Short Message

# Chapter 1

## Introduction

### 1.1 Overview

Frequent and avoidable motor vehicle crashes cause numerous deaths and injuries throughout the world. These traffic incidents may occur as a result of distractions but can be avoided if drivers had a constant awareness of other vehicles in the vicinity. Over the years, various technologies have emerged in the traffic vehicular traffic sector to help reduce accidents. Research efforts focused on Intelligent Transportation System (ITS) [5]. Vehicles that are in the ecosystem of ITS contain computing and communication modules. According to United States Department of Transportation (US.DOT), distracted driving claimed the lives of 32,166 people and injured 391,000 people in the U.S. in 2015 [13]. Additionally, According to National Highway Traffic Safety Administration (NHTSA) 37,461 people killed in motor vehicle accident in 2016 [13]. US.DOT claims that ITS can reduce or mitigate the severity of the collisions by warning drivers about potential road accidents before they occur.

Through the means of vehicular communication, the Intelligent Transportation System (ITS) aims to prevent collisions and reduce injuries by having a constant sense of awareness between vehicles. To give an example, in the event of hard braking, the only notification nearby drivers have is the red brake light of the braking vehicle.

The limitations on perception that humans have to emergency road events such as collision incidents, disruptive to normal traffic flow, might not give the driver enough time to react to avoid a collision [24]. Emerging ITS and approaches to vehicular communication are promising to reduce the propagation delay of incident reports to drives; in this regard propagation delay refers to the time from when an incident is first sensed to the time when a different vehicle would receive such a report.

Vehicular communication can occur through different means of communication; however, the shortest latency type of connection is through short to medium range wireless communication similar to WiFi known as Dedicated Short Range Communication (DSRC). In the United States, the agency responsible for regulating interstate communications known as the Federal Communication Commission (FCC), allocated 75 MHz spectrum from 5.850 to 5.925 GHz band for Dedicated Short Range Communication (DSRC) for vehicular communication formerly known as Wireless Access in Vehicular Environment (WAVE). WAVE enables vehicles to communicate status information with the aim to increase drivers' awareness to collisions [8].

## 1.2 Motivation

It is reasonable to expect that ITS and connected vehicles can have a significant impact on the safety of drivers when they are travelling; however, it is by no means a certainty. The work in this paper is motivated by the ultimate goal of improving the safety application of Vehicular Ad-Hoc Network (VANET). Many collisions incidents cause serious injuries or deaths are preventable if drivers are notified prior to the collision from occurring.

The collision avoidance system in connected vehicles depends on information encapsulated inside the Basic Safety Message (BSM), which is a type of periodic status messages emitted by vehicles. This information includes but is not limited to velocity, acceleration, direction, braking status and vehicle type. Therefore, a successful generation and delivery with a



minimized latency of BSMs is imperative for the safety application to be reliable. Different factors affect the delivery of a BSM; for example, the medium they are being broadcast within, the shadowing effect of buildings, and channel congestion, to name but a few factors.

Vehicular Ad-Hoc Networks (VANETs) suffer from channel congestion in the Medium Access Control (MAC) layer; MAC is one of the layers in the OSI stack as shown in Figure 1.1. When a vehicle is required to transmit a BSM; according to the specified transmission rate, it senses the channel and does not broadcast unless it gets a clear-to-send response from the channel. As vehicle density increases, the trend of getting a busy response from the channel increases which leads to the degradation of the safety application. Congestion control algorithms aim at increasing the scalability of the network by controlling certain transmission parameters of messages to keep the channel congestion lower.

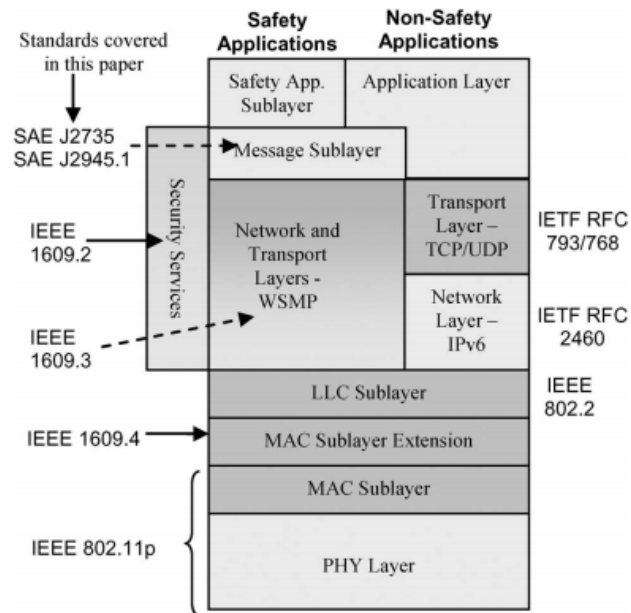


Figure 1.1: Layered architecture for DSRC communications [8]

## 1.3 Problem Statement

Vehicular Ad-Hoc Network (VANET) suffer from network channel congestion in zones with vehicle density beyond the threshold. The IEEE1609 specifies a message transmission rate of 10Hz per vehicle and a transmission range of 300 meters, in a congested stretch of highway, network congestion is imminent. In a congested network, with an overwhelming number of simultaneous transmissions of BSMS, vehicles suffer from (a) an increase in the number of packet collisions, and (b) the consequent inability to send messages reliably [11], [20].

In order that a vehicle to has enough time avoid a collision, it would need a sufficient number of status messages from surrounding vehicles in a minimal time window to make appropriate decisions [19]. Therefore, awareness is defined by the number of status messages a vehicle receives in a given time interval [10], (more on this will be discussed in Chapter 2). The time to collision required to avoid collision decreases as the speed of the vehicle increases. Therefore, it is essential to increase awareness of vehicles travelling at higher speed because they would need more space to avoid a collision. For example, a vehicle traveling with a velocity of 120 km/h requires about 200 m of space in order to avoid collision (for example coming to a complete stop) by comparison to a vehicle moving at 80 km/h would need less space to come to a complete stop[4]. It is safe to say that higher awareness is vital as vehicles gain more speed because in order to prevent a collision by slowing down or coming to a complete stop is affected by the following factors, including driver perception time, driver reaction time and deceleration time [19].

It is shown in literature packets transmitted with higher power reaches further than those with lower power [20]. Willis et al. [23] explore this approach and published an algorithm that can be improved by introducing an adaptive power level rather than having a static power level for near and far vehicles. Vehicles at a greater distance will have less awareness by virtue of less information reaching them in a given time interval; in such cases, distant vehicles would be said to have a lower priority of awareness [23].

If vehicles are transmitting at a given level while there is traffic congesting, then they are further contributing to the channel congestion in the medium they are sharing. For example, in the case of a congested stretch of highway with traffic at a standstill, an ego vehicle does not need to maintain awareness with remote vehicles at a kilometre away because their paths are not likely to intersect [23]. If we maintain an adaptive transmission power level according to vehicle's speed, then we can mitigate the channel congestion. If a vehicle is moving faster, however, then a higher power level is needed to increase awareness. Therefore, to improve channel load, different parameters have to be taken into consideration such as vehicle velocity and change in acceleration.

The broadcasting nature of the messages renders having Clear-To-Send (CTS) and/or Request-To-Send (RTS) to be disabled which further escalates the problem of channel congestion because a vehicle would not know if their messages are being received.[10].

## 1.4 Solution Outline

We have found that messages transmitted at constant high transmission power will lead to congestion and in consequence degrades the successful reception rate of packets. As a result, the amounts of packets lost over packets sent ratio is high and unacceptable for a reliable network that is supposed to save lives and prevent collisions. Furthermore, the work by authors in [23] introduce Oscillating power algorithm, in which it reduces the channel congestion by introducing transmission power patterns of alternating from low powered packets to high powered packets in an oscillating fashion. We examined the performance of their algorithm through the means of simulation using VEINS, OMNeT++ and SUMO. It is clear that while this algorithm does contribute to the performance of the network, it lacks at improving awareness with the overall vehicles in the radio sensing range.

The simulation platform we decided to use to perform this research makes it possible to collect certain parameters which are crucial to detecting channel congestion. Including but

not limited to packets sent, packets received, packets lost and total busy time by the channel sensing threshold. Other statistics are added manually that are not included in VEINS, for example, Inter-Packet Delay and fairness.

We improve on the Oscillating power algorithm by:

- Adapt the transmission power according to the vehicles speed, for example, the higher the vehicle's speed then the higher transmission power. The reason behind this approach is because vehicle's that are moving with a higher speed have less time to interact with distant vehicle.
- By gradually increasing the transmission power of packets transmitted consecutively in a cyclic fashion in order to improve awareness of vehicles in the vicinity.

Combining the two altering techniques mentioned above to the transmission power have shown an improvement in the Inter-Packet Delay and channel congestion. The outcome of this approach and results are further discussed in Chapter 4 of this paper.

## 1.5 Thesis Organization

This thesis introduces a new network congestion control algorithm for VANETs. The organization of this thesis is as follows. Chapter 2 will cover related research in the field of network congestion control in the vehicular networks. Chapter 3 will include an introduction to the new network congestion control adaptive algorithm. In Chapter 4 we present the results obtained. Finally, we give our conclusions with the recommendation for future work in Chapter 5.

# Chapter 2

## Background

### 2.1 Intelligent Transportation System

Intelligent Transportation System (ITS) is the combination of applications that target different modes of transportation and traffic management to keep a connected eco-system that is aware of each other by constant dissemination of status messages in a broadcast fashion to other vehicles. The ITS applications are not limited to safety and collision avoidance, but ITS can also be utilized but not limited to, manage traffic congestion by assigning efficient routes, toll collection, blind spot monitoring, road conditions and emergency vehicle notification [5], Figure 2.1 illustrates examples of applications that the Intelligent Transportation System (ITS) has.

The growing demand with the increasing population and having higher vehicle density on the roads tends to increase collision counts. As a result, researchers have invested in improving the scalability and reliability of ITS. There exist fundamental means of ITS to connect traffic; however, it is in the beginning stages and has to overcome challenges to become more reliable. Since having a smart inter-vehicular system relies on the dissemination and successful recipient of status messages to and from vehicles then the information

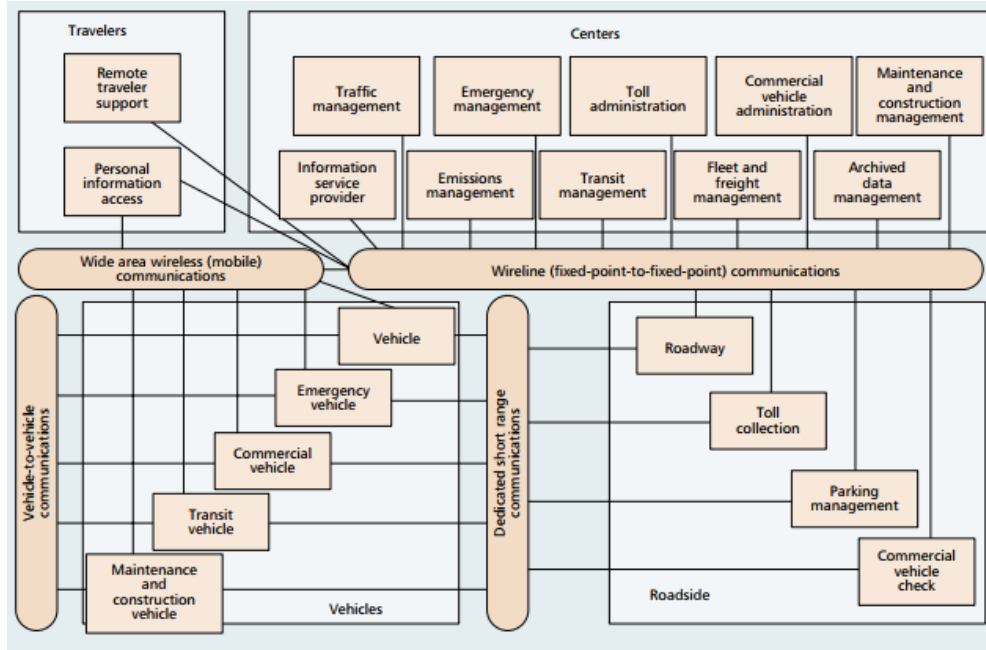


Figure 2.1: ITS Applications  
[25]

contained in these status messages has to come from a reliable source with a very minimal delay. Each vehicle contains a module known as CAN Bus which houses vital information that can be of significant usage to V2V communication. The information is encapsulated inside the module are collected from vast sensors each vehicle contains. Moreover, since the module is inside the car and does not rely on GPS to get speed and direction then there is a small latency time to get the information.

Countries have jointly come together to invest in ITS and advance its scalability. The U.S. Federal Communication Omission (FCC) dedicate 75MHz of spectrum in the 5.85 to 5.926 GHz band, as shown in Figure 2.2, that is specifically allocated for Dedicated Short Range Communication (DSRC).

In addition to the safety application of a smart transportation system, as mentioned above, it can be utilized for comfort applications. There is active research on efficient route planning that aim to decrease congestion on roads and decrease CO<sub>2</sub> emission. Comfort applications would run on a different message type known as Wave Short Message (WSM).

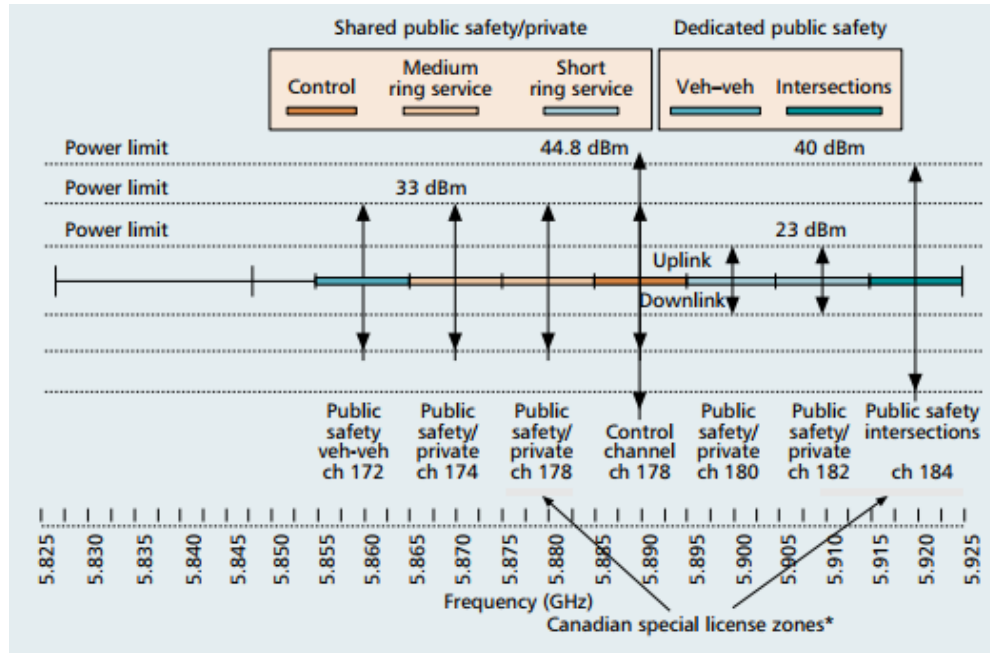


Figure 2.2: List of Frequencies dedicated for the type of message transmitted [25]

## 2.2 Vehicle-to-Vehicle and Vehicle-to-Infrastructure

A vigorous ITS is the basic foundation of reliable connected vehicle networks in order to be able to prevent collisions and save lives. A reliable and seamless connected vehicle network is a must for safety applications to be reliable and effective. To accomplish this task, vehicles have the option to communicate their status messages through different means of communication technologies such as WiFi, WiMax, LTE and Dedicated Short Range Communication (DSRC). However, studies have shown that low latency times plays a big factor for safety applications [15], [5], [8] and [7]. DSRC in the form of V2V, vehicles send packets directly to other vehicles or can send their packets to an RSU which rebroadcasts the information in a Vehicle-to-Infrastructure (V2I) fashion. An RSU can act as a relay of important information such as accident ahead or hazardous road condition.

### 2.2.1 Vehicle to Vehicle (V2V)

In this type of communication, vehicles disseminate status messages that are intercepted by other vehicles directly without the presence of other modules to relay the information. V2V communication cannot exist without an on-board module that is capable of wireless communication in the 5.9 GHz band that was allocated by the US-DOT that is capable of supporting the IEEE 802.11p standards. The network topology in this sense is very dynamic with each vehicle is considered as an independent network node. Each network node broadcast status information to nearby vehicles about GPS location, speed, direction and acceleration to inform other vehicles in the vicinity[15].

Abiding by IEEE 1609 standards, each vehicle adapt a transmission rate of 10 status messages per second. When status messages are sensed by remote vehicles, they are decrypted to extract the embedded information contained inside them followed by computation to ensure the receiving vehicle will not collide with the broadcasting vehicle. The time from the status message was created until the information was computed by the receiving vehicle have to be minimal in case if there was a collision to occur then a driver could be notified in time.

### 2.2.2 Vehicle to Infrastructure (V2I)

Vehicles can also communicate with a Road Side Unit (RSU) to relay important information for example accidents ahead, route planning and hazardous road conditions. The usage of V2I can go beyond than just simply sending status information. Researchers have shown interest in investing in this field to determine an optimal path for driving to minimize congestion and emission [14].

RSU perform in a very similar fashion to OBUs, but minimal differences such as that RSUs act as a relay rather than generate its status messages. Placement of RSU can be integrated with traffic lights, stop signs and lights. Further, the separation of each RSU on the road has to be studied for them to be able to cover a more geographic area.

Both OBU and RSU modules are capable of wireless communication with support to



IEEE 802.11. Through these modules, vehicles can exchange directly or indirectly messages that contain viable information to other drivers as outlined in SAE J2735. The information encapsulated into the messages are viable for safety and comfort applications.

### 2.2.3 Hybrid Scenario V2X

Ideally, the most stable connected network is a mixture of connected vehicles and roadside units to avoid drops in the network. In a sparsely connected system where V2V is not available, the roadside units can receive the packets and relay the information to distance that a packet cannot reach [22].

In Figure 2.3 a roadside unit acts as a relay to expand the awareness of the network. Considering the maximum transmission range of 300 meters allocated by SAE J2735 standards [8] than any distinct traffic cluster with distance more than the maximum transmission range would render vehicles in traffic clusters not aware of other vehicles in the other traffic cluster. Therefore, by having RSUs placed at strategic locations can reduce the blind spot area by connecting traffic structures together as shown in Figure 2.3 and have a more *intelligent* traffic that is more aware of other vehicles.

V2V and V2I Each communication network scenario is limited to what applications are can serve. In table 2.1, authors of [5] outline the list of applications that both V2V and V2I offer.

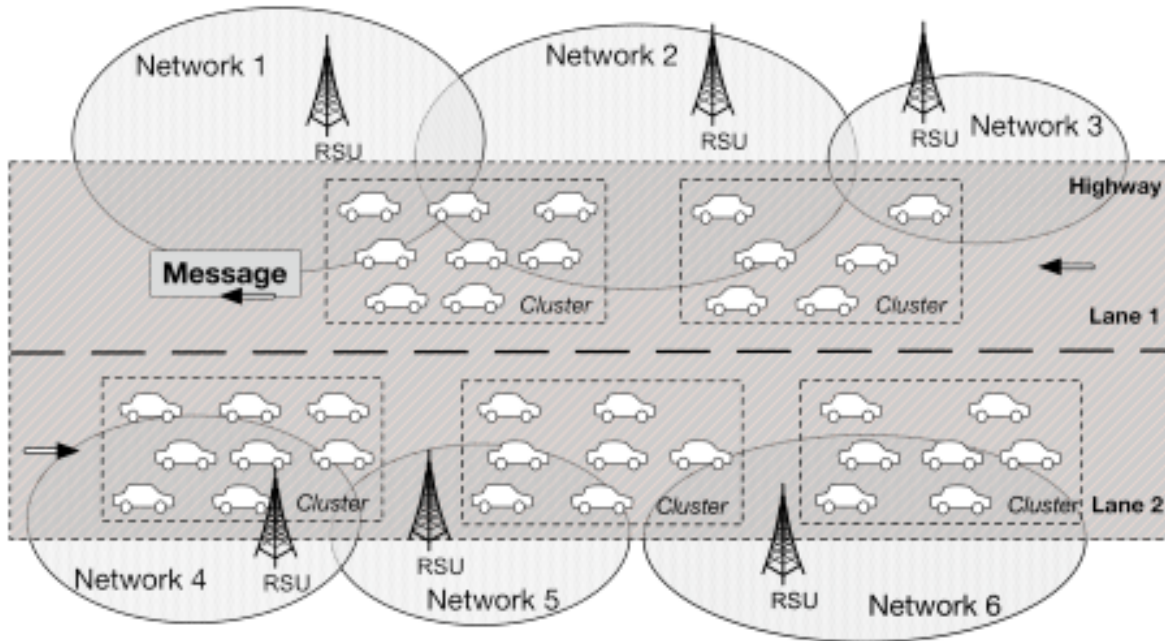


Figure 2.3: Hybrid Ad-Hoc Network: Two separated zones exchange information using an RSU. [22]

V2V	V2I
Pre-Crash Avoidance	Blind Spot Warning
Post-Crash Avoidance	Curve Speed Suggestion
Emergency Vehicle Notification	Highway/Railway notification
Blind Spot Warning	Intersection collision avoidance
Emergency Brake	Stop Sign/Traffic Light warning
Lane Change	Work Zone Warning
Road Condition	

Table 2.1: Supported applications of VANET  
[5]

## 2.3 Current Problems and Solutions

There are many challenges ITS need to overcome, such as privacy and security. In this section, we highlight the current obstacles in the network congestion control aspect of ITS.

### 2.3.1 High Node Density in MAC OSI Stack

The basis of the technology made available by using the Medium Access Control layer (MAC) for messages dissemination. MAC relies on carrier sense multiple access/collision avoidance (CSMA/CA) to avoid packet collision that relies upon the data link of the OSI stack [9]. For CSMA/CA which is depicted in Appendix D, to be able to broadcast, nodes will examine the channel's state and only transmit if the channel status is idle. Otherwise, the node will not send its packet and will wait a certain amount of random time to try to send again. However, even when using this approach, the collision rate is still not as low as it should be. This phenomenon of packet collision tends to increase as more nodes are trying to send their packets [17].

### 2.3.2 Choosing congestion control parameters

There are different reasons behind congestion in the MAC layer. To give an example, a high transmission rate or transmission power would lead to a channel congestion in a high vehicle density. Further, since *i*) vehicles cannot send an acknowledgment when receiving a message (ACK) because this would cause an ACK explosion and dramatically congest the network and *ii*) vehicles cannot request clear to send (RTS|CTS) because of the broadcast nature rather than unicast nature. Researchers have tried different approaches to reduce network congestion control in vehicles. Many of DCC algorithms use parameters such as transmission rate, transmission power, carrier sense and vehicle density in order to adjust the parameters.

### 2.3.3 Simulation

Due to the highly dynamic nature of the network, and the cost of implementing RSUs and OBUs, establishing testbeds is difficult if not infeasible. Therefore, researchers rely on the means of simulation to evaluate their envisioned ideas. To simulate connected vehicles, one would need a network simulator and a traffic simulator. In Table 2.2, we outline a list of software packages used by researchers. A common network simulator is OMNeT++. It is used to mimic network traffic by forming a network of nodes that correspond to mobile vehicles. Coupled with OMNeT++, SUMO is a traffic simulator that simulates traffic in a predefined scenario. The above two software packets are combined by using VEINS to simulate connected vehicles.

Network Simulators	Mobility Model
ns-3	SUMO
ns-2	STRAW
OMNET ++	SHIFT
JiST/SWANS	
OPNET	

Table 2.2: List of simulators used by researchers [5]

## 2.4 DSRC Challenges of interest to our Research

The vital information contained in vehicles' messages in DSRC relies on the means of broadcast rather than a point-to-point connection. Research shows that network congestion in the MAC layer increases as a result of an increase in the number of vehicle density is statistically significant; therefore, network congestion is considered one of the leading challenges against this technology. Although vehicles formulate a dynamic network of nodes, the traditional network congestion control algorithms cannot be applied here due to the highly dynamic character of the network nodes [7]

To highlight the point of the reason behind congestion control is essential in vehicular

networks, we focus on the safety application the technology offers. Collision avoidance is part of the safety application in connected vehicles. It aims to reduce avoidable accidents when drivers are not paying enough attention to the road by computing Time To Collision (TTC) with other vehicles on the road. Vehicles broadcast their status messages and are received by surrounding vehicles, when the surrounding vehicles receive the broadcast, they calculate the distance in between them. If the computation reveals the vehicles are on a collision course, then it warns the driver. Therefore, messages have to be delivered promptly for the messages to be considered able and useful.

In the United States, vehicular communication is governed by Dedicated Short Range Communication (DSRC) standards of the IEEE 1609 for Wireless Access in Vehicular Environment (WAVE). The IEEE 802.11p specifies the physical (PHY) and medium access layer (MAC). The 802.11p is an amendment to the IEEE 802.11. The European Telecommunication Standard Institute (ETSI) has similar standards for vehicular communication in Europe.

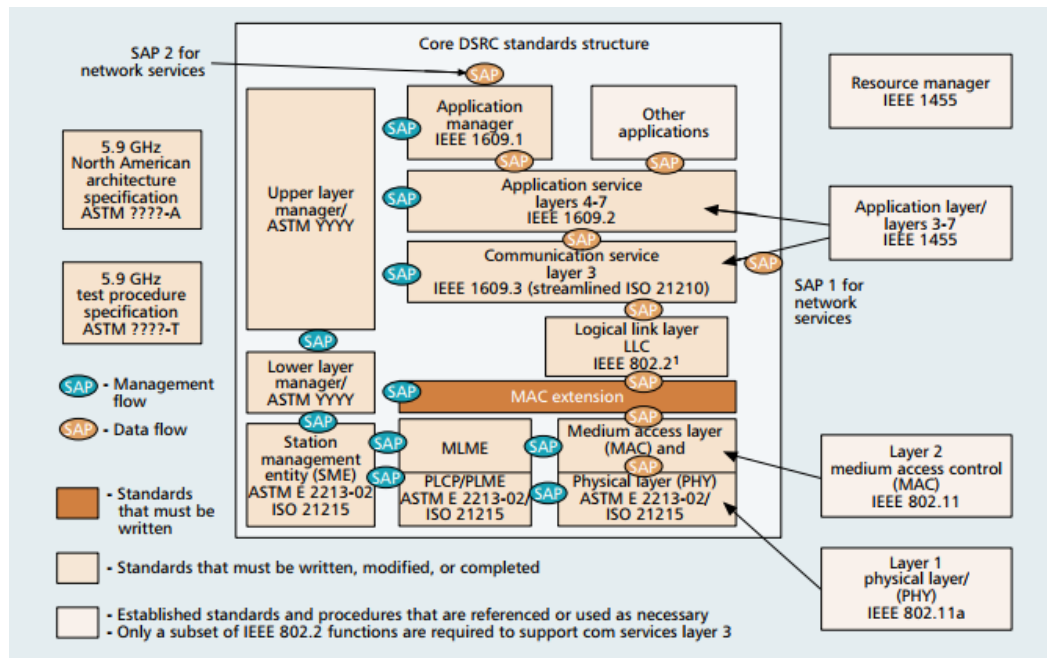


Figure 2.4: DSRC Structure [25]

## 2.5 Terminology

1. **Vehicle to Vehicle (V2V):** Term used to describe communication between two vehicles in a VANET Network. In this scenario, both participating parties have to be equipped with a wireless capable module known as an On-Board-Unit (OBU) with support to IEEE 802.11 [8].
2. **Vehicle to Infrastructure (V2I):** Similar to V2V but in this case vehicles relay their messages to a Road-Side-Unit (RSU) that is capable of wireless communication and has to be able to support IEEE 802.11 [8].
3. **Vehicular Ad-hoc Network (VANET):** It is the spontaneous creation of a wireless network consists of highly dynamic mobile nodes to exchange information for traffic safety and/or comfort [8].
4. **Wireless Access in Vehicular Environment (WAVE):** Made available by the enhancement to the IEEE 802.11 by amending "p" becoming 802.11p which is required by ITS. This amendment makes it feasible for the data exchange between high-speed mobile nodes (vehicles) and RSU in the designated 5.9 GHz band allocated for vehicular telecommunication [8].
5. **Dedicated Short Range Communication (DSRC):** Is a medium range wireless communication that is utilized by ITS to run its safety and comfort applications. It permits very high data transmission which is critical for ITS. The United States Department Of Transportation US.DOT has allocated 75Mhz of spectrum in the 5.85 to 5.925 GHz band for the vehicular communication [8].
6. **Cooperative Awareness Message (CAM):** The counterpart of the North American DSRC BSM is CAM. The structure is comparable to the BSM, but with few differences,

for example, hazardous driven messages are included in a separate message known as Decentralized Environmental Notification Message (DENM), while in DSRC such events are included in the BSM. The structure of CAM is outlined in ETSI TS 102 637-2 which is generated by the CAM Management. ETSI also adopts IEEE 802.11p to the European spectrum with a requirement of having a decentralized congestion control management to avoid channel congestion while in the North American, network congestion control is not required [8].

7. **Basic Safety Message (BSM):** The type of message that encapsulates information outlined in the SAE J2735. The message is the basis of ITS, which safety and comfort applications use. It is composed of two parts, Part I and Part II, table 2.3 outlines the BSM structure. The first part includes information that is critical to vehicles safety in the VANET environment and the second part includes information that relates to the comfort application for vehicles in the VANET environment. These messages are transmitted with a 10Hz rate in order to maintain awareness for ITS. The current network congestion control in VANET works on improving a successful transmission of the message promptly for it to be considered usable [8].

BSM blob I	BSM blob II
MsgCnt	Vehicle Event Flags
TemporatyID	Path History
DSecond	Path Prediction
Latitude	Exterior Lights
Longitude	
Elevation	
PositionalAccuracy	
TransmissionAndSpeed	
AccelerationSet4Way	
BrakeSystemStatus	
VehicleSize	

Table 2.3: BSM Part I and II Structure<sup>1</sup>.

## 2.6 Algorithm Performance Analysis

Network congestion control algorithm aims to reduce channel load and increase the throughput of the network to maintain a reliable service. However, there are different approaches to handle network congestion. To evaluate each method, there are specific parameters of interest to researchers that are used to evaluate an algorithm. The software packet, VEINS that couples the network and traffic simulator comes equipped with parameters that are useful for algorithm evaluation. Other, more specific parameters have to be implemented outside of VEINS for further evaluate algorithms such as algorithm fairness and Inter-Packet Delay.

1. **Packets Sent:** Consists of all the packets transmitted by a node to the network medium that is of type BasicSafetyMessages, WaveShortMessage and WaveShortAdvertisement. The number of packets sent is useful when comparing algorithms that adopt the different technique to handle congestion. For example, the packets sent would be different when a rate control algorithm is used over a power control algorithm.
2. **Packets Received:** This is the number of packets received by a vehicle regardless of its type. It would give us an indication of how many packets lost when compared the total packets sent. Essentially, we would want this number to be higher as much as possible to increase awareness of connected vehicles. Therefore researchers tweak their algorithms to increase its value.
3. **Channel Busy Ratio:** The ratio of the response a busy over the clear-to-send a vehicle receives when it opens a channel and request to send. This ratio tends to be higher in a congested network. Changes in CBR ratio is a clear indication of how the algorithm is performing on keeping the channel congestion under acceptable level. Most often, it is used as an input parameter to many network congestion algorithms.



4. **Inter-Packet Delay:** The time gap between two consecutive packets a vehicle receives from a neighbouring vehicle. A lower IPD number indicates that vehicles are better aware of each other because there are more position updates. The IPD increases as the network gets more congested because vehicles cannot send their packets or packets collisions occur.
5. **Fairness:** Algorithm fairness is used to evaluate congestion control algorithms to determine a shared use of resources available to all nodes within the medium they are sharing. Jain's fairness index is widely used to evaluate TCP congestion control algorithms in regular networks. The same tool can be used to measure algorithm fairness in vehicular network congestion control [12].

## 2.7 Literature Review

From the algorithms that target network congestion control, the authors of [23] created a novel algorithm to handle the problem by altering the amount of transmission power. The algorithm exploits the relationship between the transmission power of packets and the distance a packet travels. Rather than tackling network congestion to a parameter used to measure channel congestion, their approach addresses the problem before congestion occurs. The authors distinguish near and far vehicles by having two different rate and power transmissions as  $Tx_n$  and  $Tx_f$  as near and far respectively. The authors note by doing so; they would be sacrificing awareness for distant vehicles while maintaining higher awareness for closer ones. This approach is acceptable because drivers have less reaction time to avoid accidents with closer vehicles than distant vehicles. While this algorithm improves the number of packets received and reduces the number of packets lost, it could be further improved by introducing more transmission power patterns.

Another approach in [12] where authors design a Packet count decentralized data-rate congestion control algorithm (PDR-DCC). The algorithm computes the number of pack-

ets vehicles received and distribute a homogeneous distribution of data-rate among all vehicles then converge to a global CBR. Both approaches by [23] and [12] aim for channel usage to be under a certain threshold, but the difference is that the input parameters the algorithms use.

As mentioned above that CBR is a standard input parameter for network congestion control. Authors in [9] use CBR computed from each vehicle then determine a global CBR threshold followed by adjustment of transmission rate to maintain a CBR ratio below the computed global CBR. The same authors from [9] further improved the algorithm, therefore, creating Error Model-based Adaptive Rate Control (EMBARC) [2]. Initially, the authors adjust the transmission rate to maintain a channel load under a certain threshold by having each vehicle contribute information creating a global channel busy ratio. The enhancement to the algorithm in EMBARC has introduced another parameter which they formally define as *Tracking Error* TE, to trigger transmission when an error is detected using vehicle kinematics.

Authors in [10] use a random transmission power control (RTPC) to adjust channel congestion. If vehicle transmits with a uniform power and transmission rate, then the channel congestion noted is also uniform and consistent with the transmission parameters. Furthermore, the authors state that concurrent packet collision affects the awareness level of the system significantly due to the continuous non-reception of status messages from vehicles rendering the last one received to be outdated. However, if packet drop is between consecutive packets rather than random packets, then the awareness level will drop, but it will not be as significant to be below a threshold. Therefore, RTPC aims to randomize the transmission power to have a heterogeneous packet to a collision. This approach has demonstrated improvement in CBR ratio but at the cost of vehicle awareness.

Another approach is to correlate the transmission power and transmission rate by finding optimal transmission parameters as the authors of [20] have described. The algorithm they designed uses the transmit range required at an instant then computes the transmit

power required followed by mapping the transmit rate to be under a certain threshold of channel congestion. The authors evaluate their algorithm based on the number of packets received within a safety range, which they formally define as *Safety ThroughPut*. This approach demonstrates improvement in channel congestion but like mentioned above due to the unpredictability of the randomness nature of radio propagation, this approach might not be feasible in real-world scenario.

Authors in [18] derive two approaches to mitigate channel congestion. The first method is to have three states of channel congestion measured by channel load (CL). For example, if a CL falls between specified target, then the transmission parameters are adjusted accordingly until the CL falls under the specified CL for that state. The algorithm keeps awareness of the CL of all time and determines to switch to a different state if conditions are satisfied. A second approach the same authors tried is a *synchronous* approach, meaning the transmission interval is divided into time slots where vehicles can inject their information without overlapping with other vehicles also broadcasting.

Table 2.4 summarizes the approaches and outlines simulations platforms researchers have used to handle network channel congestion.

Table 2.4: Classifications of Congestion Control Algorithms

<b>Paper</b>	<b>Control Parameter</b>	<b>Performance Metrics</b>	<b>Simulation Scenario</b>	<b>Network Simulator</b>	<b>Traffic model</b>
Fair Decentralized Data-Rate Congestion Control for V2V Communications (2017) [12]	Data Rate	CBR, Packet Count	Highway	ns-3	SUMO
Random transmit power control for DSRC and its application to cooperative safety (2016) [10]	TX Power	CBR, Packets Lost, IPD	Highway, Urban	Not Documented	Not Documented
Decentralized Congestion Control Algorithm for Vehicle to Vehicle Networks Using Oscillating Transmission Power (2015) [23]	TX Power	CBR, IPD, BER	Highway	OMNET++	SUMO
Joint Space-Division Multiple Access and Adaptive Rate Control for Basic Safety Messages in VANETs (2014) [20]	TX Power, Transmission Rate	Safety throughput	Highway	Not Documented	Not Documented
EMBARC: Error Model based Adaptive Rate Control for Vehicle-to-Vehicle Communications (2014) [2]	Transmission Rate	TE	Highway	ns-2	SUMO
LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control (2013) [9]	Transmission Rate	CBR	Not Documented	ns-2	Not Documented
Congestion Control for Vehicular Safety: Synchronous and Asynchronous MAC Algorithms (2012) [18]	TX Power	Channel Load	Highway	ns-2	Not Documented

## Chapter 3

# Proposed Power Adaptation algorithm

### 3.1 Introduction

Vehicles that are part of the Intelligent Transportation Network (ITS) can communicate using a (V2V) or a (V2I) communication type. In both types of communication, a reliable inter vehicular network is a must to maintain the reliability of the ITS applications.

Research shows the reliability of network to broadcast packets and successful transmission declines as network density inclines. The 802.11 have shown in previous studies mentioned in Chapter 2 that with higher vehicular densities, it suffers from congestion. From a network perspective, it means, the safety messages are not delivered on time or maybe not delivered at all. The information Intelligent Transportation System (ITS) needs to function is already available through each vehicle's CAN Bus module. The difficulties in the research are the successful and timely delivery of this information. In our research, we observe the limitation of the network utilizing simulation using OMNeT++ because gathering real-time data is unachievable due to the unavailability of OBU's inside vehicles.

In this chapter, we present a novel congestion control algorithm for vehicular networks. The network congestion control algorithm proposed in this thesis alters the transmission power in order to improve the rapid increase of IPD in the oscillating power algorithm.

Adjusting the transmission power would dictate the distance a packet travels. Therefore, any application that alters the transmission power to reduce congestion is substantially adjusting the travel distance of packets. Such a way to alter transmission power is helpful to reduce congestion control because vehicles with a close approximation of each other would experience less packet pollution from distant vehicles which have no immediate relevance to the vehicles' safety.

## 3.2 Synopsis of Network Congestion Control

A perfect network would mean all the packets transmitted are successfully received by recipients; in other words, zero percent loss of packets, however, this is not achievable, in practice. The current research mentioned in Chapter 2 aims to improve the network throughput but each approach has its own tradeoff.

Different means can be used to assess the performance of congestion control algorithm. To give an example, Channel Busy Ratio (CBR), Inter-Packet Delay (IPD) and but not limited to Beacon Receive Rate (BRR) [23]. Researchers in the field have developed algorithms to handle network congestion control using CBR as a means of assessing network congestion, for example, the algorithms developed by [2] and [1]. While both of these approaches use a novel method of controlling channel load, however, these approaches address the network congestion problem when the channel load is increasing. On the contrary, algorithms that do not use a measure of channel congestion parameter, aim to keep the channel load as low as possible by altering transmission range and transmission frequency as a tool to avoid network congestion regardless of what the channel load is.

Vehicles become aware of each other through the exchange of status messages; hence the more status messages, the more vehicles are aware of each other and less likely to collide. **Awareness** is formally defined by [10] as: *It is the ability of an application to know the status, e.g. position, speed, heading, of neighbouring vehicles. Awareness is qualified by*

*its range, i.e. distance at which the application at most becomes aware of vehicles, and its quality, i.e., accuracy/up-to-dateness of the status information.*

### 3.3 Architecture of proposed algorithm

The proposed algorithm differs from other algorithms in this field by a) The continuous expansion of awareness circle b) adjust transmission according to vehicle's speed. The research studied and discussed in chapter 2 choose a parameter to detect congestion and act upon it, for example, CBR. Authors in [23] alters the transmission power to decrease channel congestion by sending a number of low powered packets to intentionally to reach close range followed by a sufficiently powered packet to maintain awareness for distant vehicles. While the algorithm shows improvement to channel congestion, however, awareness is sacrificed. Further, the transmission power was chosen for nearby vehicles and for distant vehicles are fixed levels. We improve on this algorithm by raising the transmission power gradually to increase the awareness circle followed by a high powered packet to maintain awareness of distant vehicles. Besides, the transmission power in the proposed algorithm is also increased in vehicles with higher velocity to raise awareness with distant vehicles because with a higher speed, the Time-To-Collision decrease (TTC).

The proposed algorithm tries to achieve the following:

- Keep channel congestion as low as possible.
- Improve packets received over packets lost.
- Improve awareness by reducing the Inter-Packet Delay (IPD).

#### 3.3.1 SpeedFactor

The reason to select a speed factor to give vehicles a higher transmission power is to reduce channel congestion on a stretch of highway when vehicles are not moving as fast.

For two vehicles that are travelling on the same road with different velocities, the vehicle with the higher velocity will cover more distance than the slower vehicle and would encounter with more vehicles along the line that the slower vehicle has yet to encounter as shown in Figure 3.1. Therefore, if we maintain lower transmission power for the slower moving vehicle, we will be reducing channel congestion by virtue of reducing the transmission power level and in return would have packets travelling a shorter distance.

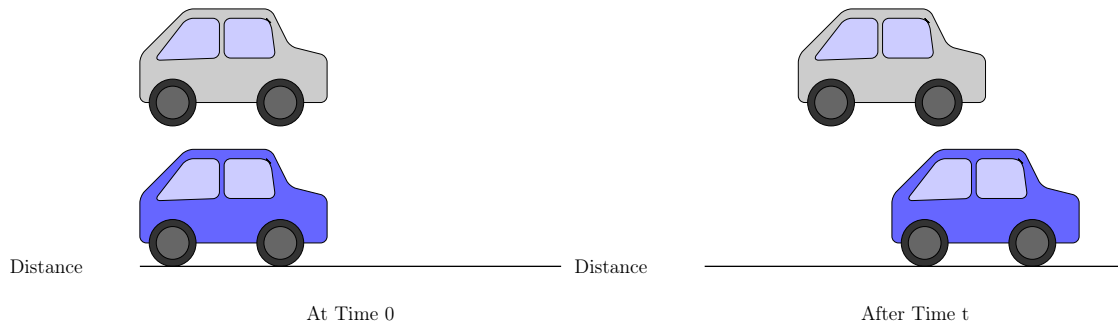


Figure 3.1: Vehicles with different speeds

### 3.3.2 Expansion of Awareness

We choose to increase the power of transmission in a cycle fashion. In other words, increase the power of transmission for each consecutive packets under a predefined cycle length. Once the number of consecutively transmitted packets reaches the cycle length, then the power of transmission is set to the lowest as shown in Figure 3.2. Using this approach, we would sacrifice awareness for distant vehicles in order to reduce the channel load and reduce the number of packets lost.

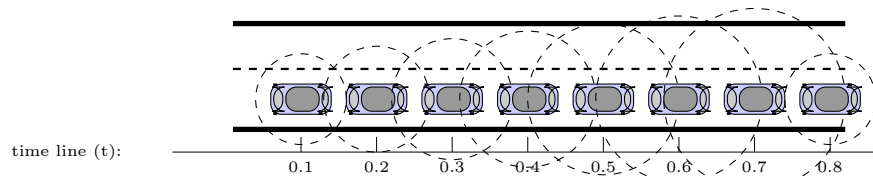


Figure 3.2: Increasing power of transmission for consecutive packets



### 3.4 High Level Outline

The number of transmissions chosen until the transmission power is reset back down to lowest is determined by each vehicle. Each vehicle keeps count of how many packets it sent in a local variable in the application class to create a cycle. The cycle is reset when the counter variable reaches a limit and then the counter is set back to one. The counter is used to set the transmission power in the Medium Access Control (MAC) module. The speed of vehicle factor is determined based on the vehicle's speed and then multiplied by the CycleCounter, therefore, vehicles moving with a higher velocity would have a higher transmission power level, given they are at the same CycleCounter.

---

```

1: procedure GETSPEEDFACTOR
2:   VehicleSpeed = getVehicleSpeed()
3:   if (VehicleSpeed > 90.0) then return SpeedFactor ← Max
4:   else if ((60.0 < VehicleSpeed ≤ 90.0)) then return SpeedFactor ← High
5:   else if ((40.0 < VehicleSpeed ≤ 60.0)) then return SpeedFactor ← Medium
6:   else if ((vehicleSpeed ≤ 40.0)) then return SpeedFactor ← Low
7:   end if
8: end procedure
9: procedure ALLOCATE TRANSMISSION POWER LEVEL
10:  if (CycleCounter = CycleLength) then
11:    CycleCounter ← 1
12:    SetTxPower(MaxTxPower)
13:  else if (CycleCounter <> CycleLength) then
14:    SetTxPower(CycleCounter * SpeedFactor)
15:    CycleCounter = CycleCounter + 1
16:  end if
17: end procedure

```

---

Figure 3.3: Adaptive Algorithm

The adaptive algorithm is invoked continuously as a vehicle sending packets regardless of the vehicle speed. With a constant transmission rate of 10 Hz, the algorithm is called ten times per second. However, it might differ if the algorithm would have been altering the transmission rate. Both of the maximum transmission power and CycleLength

are set initially in the initialize method for every vehicle. The initialize method is invoked when a vehicle is created in the simulation environment.

Following the pseudocode in Figure 3.3, to determine the transmission power, the algorithm is being divided into two parts. The first part, it determines the speed factor and the second part determines the Transmission power. The speed factor is a variable that increases as the vehicle's speed increases. We identify four-speed factors, all of which increase in weight respectively such as; Low, Medium, High, and Max speed factors. If the speed of the vehicle is less than or equal to 40.0 km/h then we select Low. The second speed factor is selected when the speed of the vehicle is greater than 40.0 km/h but less than or equal to 60.0 km/h. The third speed factor is selected when the speed of the vehicle is greater than 60.0 km/h but less than or equal to 90.0 km/h. Lastly, the higher speed factor is selected when the speed of the vehicle is more than 90.0 km/h. Moreover, a restriction when selecting a speed factor value is that the predecessor cannot be of greater weight than its successor.

The second part of the algorithm is selecting the Cycle Length. Our approach is to expand the awareness circle by increasing the transmission power level after consecutive transmissions. A continuous expansion of awareness circle is not possible because it would dramatically congest the channel. Using a cycle length would allow us to reset the transmission power to a lower value than increase it gradually again. The CycleCounter is incremented every time a vehicle sends a message. Once the CycleCounter has reached the CycleLength, then at that point, the message is transmitted with the maximum transmission power level to reach far distance to maintain awareness with distant vehicles, followed by a reset of the CycleCounter count.

The final step of the algorithm is the CycleCounter is multiplied by the Speedfactor that was computed in the first part of the algorithm to produce a final transmission power for the vehicle to use to transmit its packet in a specific instant. The steps of the algorithm is formally presented in pseudocode in Figure 3.3 and a more detailed flowchart in Figures 3.5,3.6 and 3.7.

Iteration	Speed < 40.0 km/h	40.0 km/h < Speed ≤ 60.0 km/h	60.0 km/h < Speed ≤ 90.0 km/h	Speed ≤ 90.0 km/h
1	1 * Low	1 * Medium	1 * High	1 * Max
2	2 * Low	2 * Medium	2 * High	2 * Max
3	3 * Low	3 * Medium	3 * High	3 * Max
4	4 * Low	4 * Medium	4 * High	4 * Max
5	5 * Low	5 * Medium	5 * High	5 * Max
6	6 * Low	6 * Medium	6 * High	6 * Max
7	Max Power	Max Power	Max Power	Max Power

Table 3.1: Transmission Powers chosen for each vehicle

Incorporating the speed of the vehicle as a first input parameter to the network congestion control algorithm and a consecutive increase of the transmission power as a second input, we would have different allocations of transmission powers. Table 3.1 outlines the convergence of the transmission power allocation the adaptive algorithm uses. We believe this would mitigate the network congestion control by bringing the channel usage to a lower ratio and improvement on the Inter-Packet Delay (IPD).

The desired transmission patterns of packets transmitted without taking into account the speed factor are demonstrated in Figure 3.2. As it is illustrated, after each broadcast packet, the following packet is sent with a higher power to expand the awareness circle.

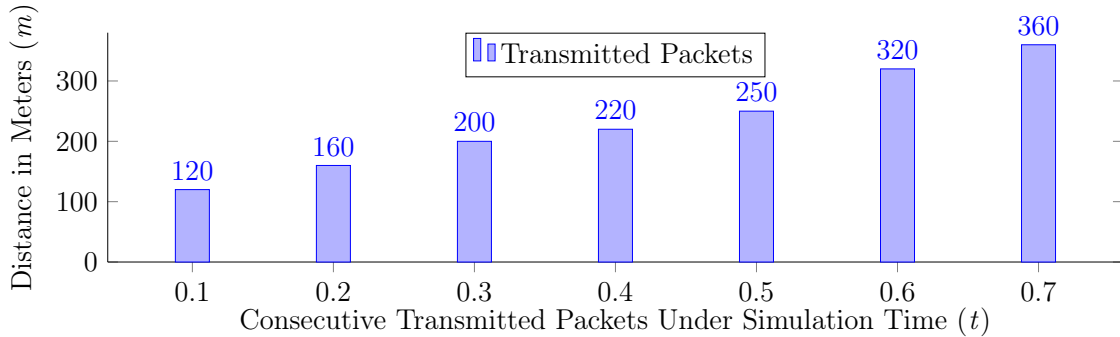


Figure 3.4: Desired Travel distance of transmitted packets

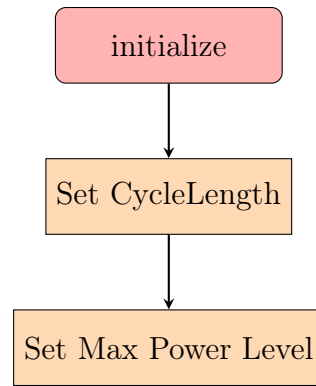


Figure 3.5: Initialize method called at vehicle creation

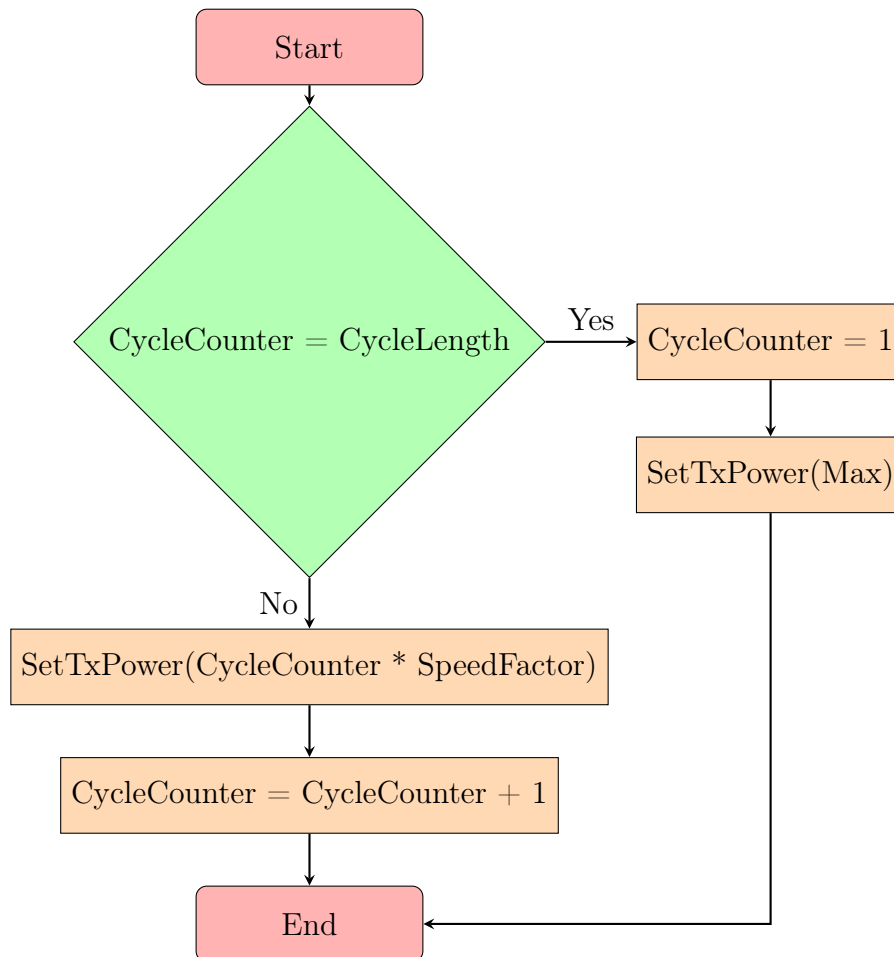
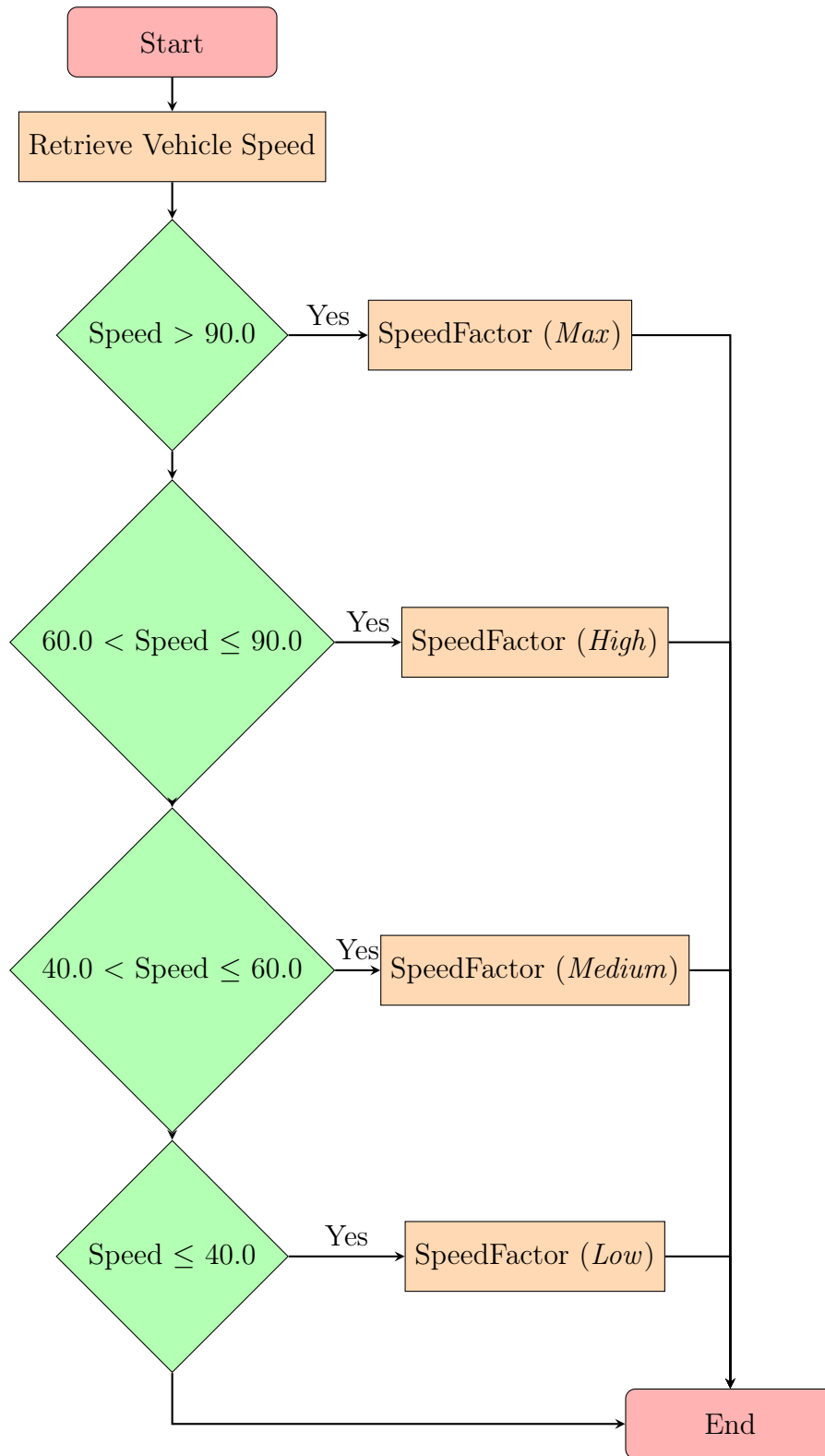


Figure 3.6: Adaptive method called before every packet is transmitted. Procedure 2.

Figure 3.7: Adaptive method called before every packet is transmitted. Procedure 1.



# Chapter 4

## Experiment and Results

### 4.1 Simulation

To simulate our envisioned work, we use a collection of open source software to do so. In a vehicular communication network can be thought of, part as a collection of mobile nodes that exist within close proximity of each other and communicate via means of broadcasting as network nodes, and another is the simulation vehicle traffic scenario. Therefore to simulate vehicular traffic: a) we need to simulate vehicle traffic b) simulate network communication between vehicles. We model the network simulation using OMNeT++ which is a well-established network simulation environment based on C++ [21]. For the latter case, to simulate traffic scenario, we used vehicle traffic simulator known as Simulation of Urban MObility (SUMO) [3], which is an open source framework that is used by large research projects. Lastly, to connect SUMO and OMNeT++, we used an open source package Vehicles In Network Simulation (VEINS) couples the network and traffic simulator by exchanging information through a local socket [16].

In the process of evaluating the network congestion control algorithm, we focus on transmitting BasicSafetyMessages (BSMs) only; we did not broadcast any other type of messages. For each simulation we recorded the messages received, fields of interest contained

inside each message are:

- Creation Time
- Received Time
- Sender ID
- Sender coordinates
- Sender Speed

Parameter	Value
Data Rate	6Mbps
sensitivity	-89dBm
thermalNoise	-110dBm
Transmission Rate	10Hz
BasicSafetyMessage size	250Bytes
Simulation Duration	100s

Table 4.1: Simulation Parameters

### 4.1.1 Simulation Setup

Each simulation we conducted consist of its scenario files that dictate how the simulation behaves. For example the creation of each node, the frequency of creation of nodes, a behaviour of nodes and the roads to simulate and speed of node. The network simulator, OMNeT++ uses a configuration model files (*ini*, *ned*) to specify simulation parameters. Table 4.1 outlines the the parameters used for our simulation runs that were added to the *ini* file of each scenario as shown in Appendix A.

Vehicles were added every 0.01 seconds to the simulation environment when possible without colliding with another vehicle by SUMO route configuration file. Once a vehicle is created in SUMO, then the *sumo-launchd-py*, which is a daemon that runs in the background and constantly listening to incoming requests, creates a corresponding network node on the OMNeT++ side for network communication. Vehicles are constructed with a maximum speed of 100 km/h, and they enter the simulation by picking a lane randomly, and they maintain their path until the end of the road. Once a car reaches the end of the road, then

its job is done and no longer transmits or contributes to the network followed by execution of *finish()* function; which is used for data collection.

Vehicles travel route predefined in the SUMO configuration file. The flow of traffic is defined by route id in the *rou* file, where each vehicle flows from one road edge to another road edge. The route file contains parameters the vehicles will exhibit, such as maximum speed, acceleration and deceleration. For detailed SUMO configuration files, please refer to Appendix B.

### 4.1.2 Simulation Runs

We created a total of four scenarios to evaluate the different approaches to handle network congestion control. The map configuration of the highways of I-75 and E.C. Row where downloaded from OpenStreetMap (OSM). OSM is a map of the world that is designed and constantly updated by a group of volunteers that are dedicated to creating an open source software [6].

Post a simulation start, the network simulator OMNeT++ creates network nodes to simulate network traffic, regardless if they are stationary or mobile. Figure 4.1 is a visual representation of the mobile nodes OMNeT++ creates while simulating a 12 lane scenario.

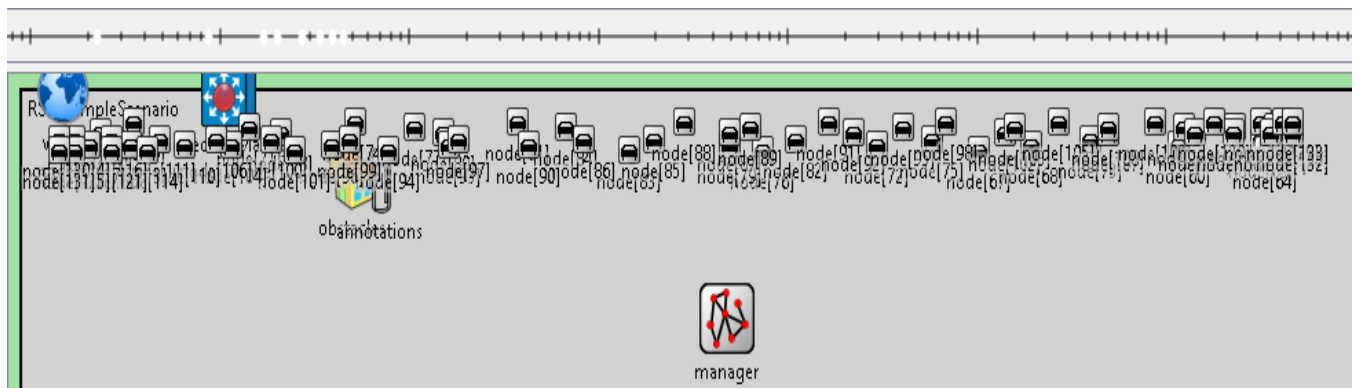


Figure 4.1: Mobile OMNeT++ Nodes Representation



The four scenarios we tested our algorithm with are:

- Scenario of six-lane highway (Three lanes each direction)
- Scenario to simulate on Interstate I75 in Detroit Michigan U.S.A
- Scenario to simulate on Edward Charles Row (E.C Row) In Windsor, Ontario Canada
- Scenario of twelve lane highway (Six lanes each direction) - To stress the network



Figure 4.2: Scenario: Detroit, Michigan, U.S.



Figure 4.3: Scenario: Windsor, Ontario, Canada

Each run was simulated for 100 seconds and with a transmission rate of 10 Hz. The cycle length we chose for our algorithm to run was seven cycles. Therefore the transmission power would start at one mW multiplied by the speed factor until the cycle counter reaches the cycle length where we broadcast at 10 mW. We performed a total of three simulations per scenario each of which we apply different network congestion control algorithms. First run was without the use of a network congestion control algorithm. We used the oscillating power adaption to perform the second run. The adaptive power adaption algorithm was used to perform the third simulation.

The first task of the algorithm is to allocate a speed factor to be used to increase the transmission power depending on the vehicle's speed. Four different speed factors were chosen, low, medium, high and maximum. The low-speed factor was given a weight of 1.05 for vehicles travelling 0 to 40 km/h. The Medium speed factor was given a weight of 1.1

for vehicles travelling with a speed of 40.0 to 60.0 km/h. The high-speed factor was given a weight of 1.2 for vehicles travelling with a speed of 60.0 to 90.0 km/h. The high-speed factor was allocated a weight of 1.4 for vehicles travelling with a speed of more than 90 km/h. Speed factor allocations are demonstrated in table 4.2.

<b>SpeedFactor</b>	<b>Speed Range (km/h)</b>	<b>Value</b>
Low	0 - 40	1.05
Medium	40 - 60	1.1
High	60-90	1.2
Maximum	> 90	1.4

Table 4.2: Speed Factor Assignment

## 4.2 Results

In this section, we explore the results collected from the four different approaches of handling congestion control in Dedicated Short Range Communication (DSRC). First, we model of how congestion affects vehicular communication without using a congestion control approach. Followed by modelling of the Oscillating power approach. Lastly, we model our adaptive approach of controlling the transmission power.

The four scenarios we chose to simulate varies by the number of lanes, consisting of two lanes to six lanes each direction. Since the simulator populates every path independently, then we would expect a different number of cars for each scenario. Table 4.3 outlines the number of vehicles we observed in each scenario. The number of cars does not change when modelling the different approaches mentioned in the above paragraph.

Scenario	Number of Vehicles
Six Lanes	342
12 Lanes	691
E.C. Row	198
i75	307

Table 4.3: Number of vehicles observed

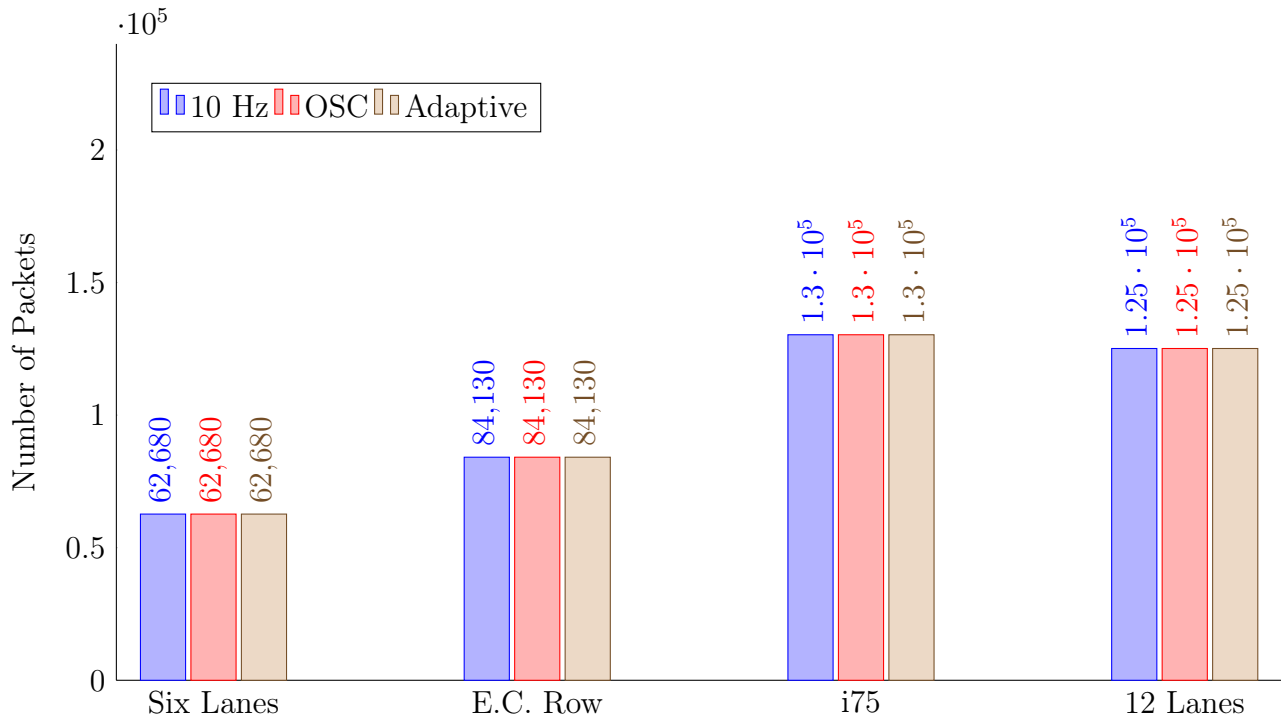


Figure 4.4: Total Sent Packets

### 4.2.1 Packets Sent

First we examine the total numbers of packets sent by vehicles in each scenario separately. Figure 4.4 shows the number of packets sent each scenario grouped together with three multiple runs. It is clear that regardless of the chosen algorithm, we experience the same number of total packets. That is expected because we chose a simulation rate of 10 Hz. Further, since the algorithms used are power adaption algorithms, therefore they do not alter the transmission rate. Regardless of algorithm was used, vehicles will always send the same number of packets.

## 4.2.2 Packets Received

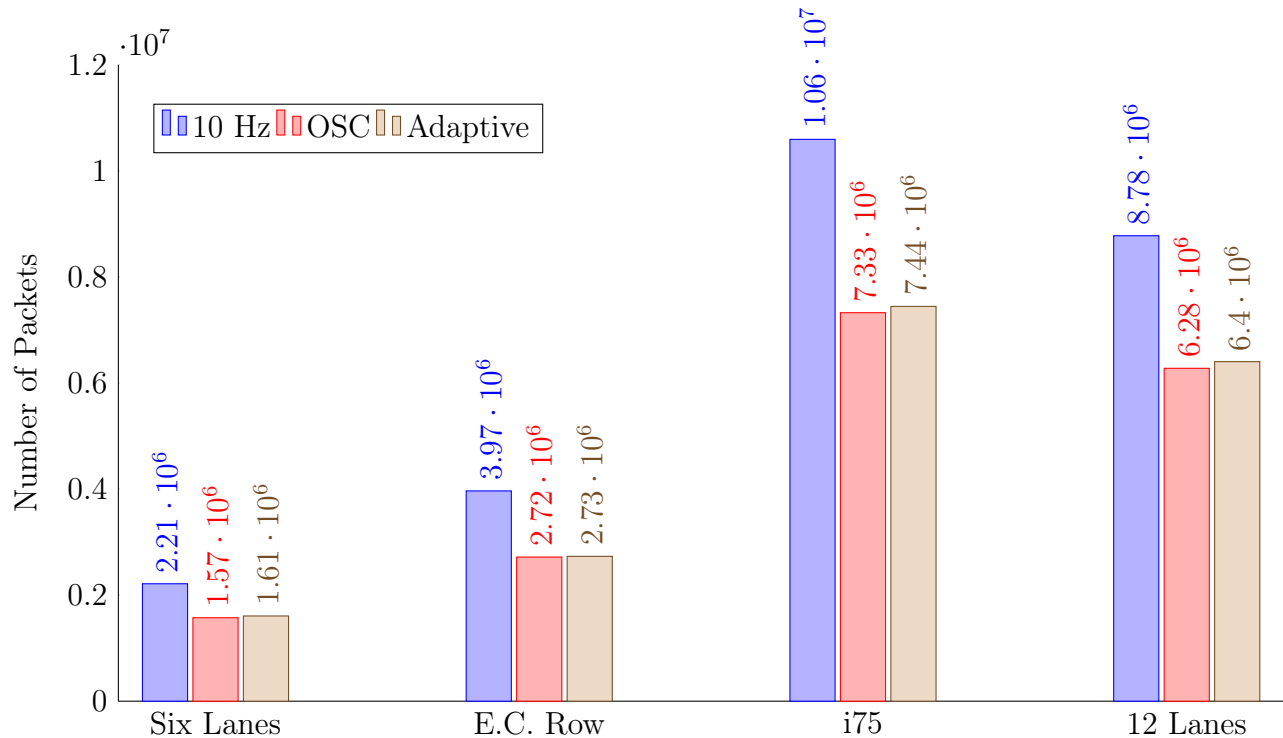


Figure 4.5: Total Received Packets

The number of received packets is Dependant on the congestion control algorithm. The number of received packets by itself, does not dictate the quality of the algorithm. Therefore, the higher number of received packets does not necessarily mean that algorithm with more received packets means it is a better algorithm. Since both algorithm exploit the the distance traveled by a packet and the power of transmission. Therefore, we would expect to have less number of packets received while using OSC and Adaptive algorithm compared to running the simulation without a network congestion control algorithm. The shorter travel of packets is illustrated in Figure 4.5 by observing both OSC and Adaptive to have lower received packets due to the shorter distance of travel of packets.

Simulating without a network congestion control algorithm results in receiving more packets. That is expected because since the packets are always transmitted with full power, then they will reach further distances. Vehicles at a range will receive packets from

further vehicles but this will not contribute to improving the safety application of vehicles because this type of maximum power transmission contributes to congesting the channel.

### 4.2.3 Lost Packets

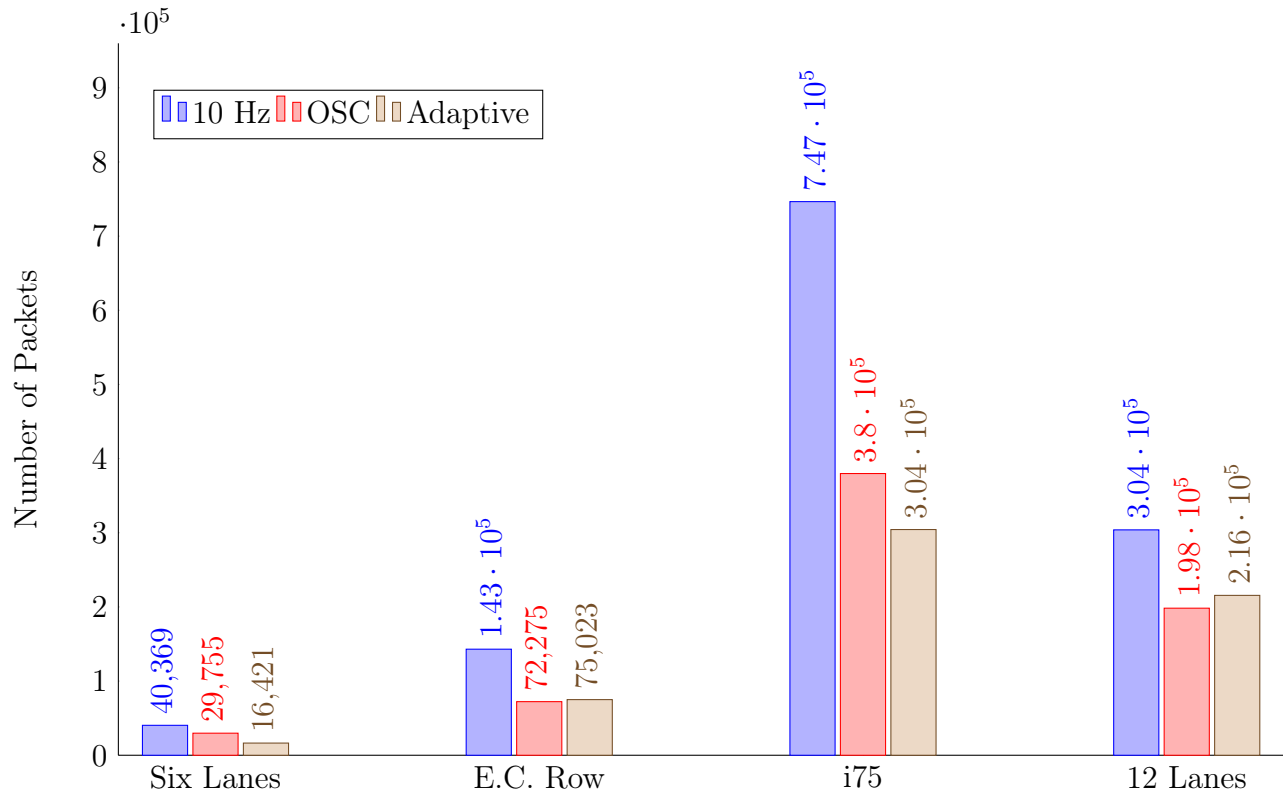


Figure 4.6: Total Lost Packets

The goal to improve the safety application is to reduce the amount of packets lost. We notice in Figure 4.6 that the number of packets lost when simulating without using a network congestion control algorithm is higher we experience a significant loss of packets. Both OSC and Adaptive network congestion control algorithms reduce the amount of packets lost and both perform similarly in reducing the amount of packets lost.

Both of OSC and Adaptive adapt a transmission power in different ways, but the performance difference of OSC and Adaptive in terms number of packets lost is similar to each other with minimal variation. Our goal to improve the safety applications by reducing

the rapid increase of Inter-Packet Delay (IPD) in vehicles while maintaining a similar loss of packets between OSC and Adaptive algorithm.

#### 4.2.4 Beacon Receive Rate (BRR)

Comparing the ratio of packets received over packets sent, we notice the Adaptive algorithm performs slightly better than OSC by having a better reception rate as shown in Figure 4.7. Simulating without a network congestion control demonstrate to have a higher reception rate compared to the network control algorithms because both OSC and Adaptive algorithms aim to reduce the reception of packets for distant vehicles. Therefore by having packets travelling longer distance, increases the reception rate for distant vehicles but the improvement it has for the safety application is negligible. Furthermore, by having packets travel longer as shown in this case, it further contributes to the congestion of the channel and results to the degradation of the safety applications and directly increases the Packet Error Rate, more on this will be discussed in section 4.2.5.

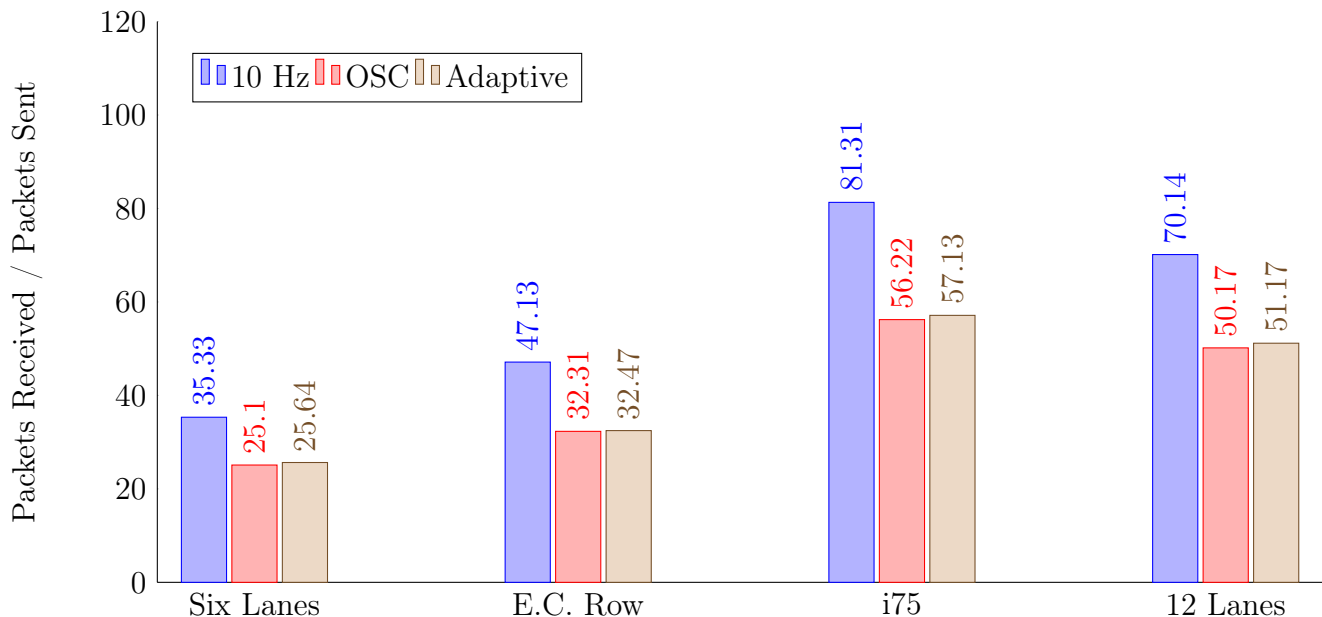


Figure 4.7: Beacon Receive Rate

## 4.2.5 Beacon Error Rate (BER)

Having a lower beacon error rates would have a positive impact on the safety application. Figure 4.8 Shows the performance of both OSC and Adaptive with Adaptive being slightly better at reducing the beacon error rate over the OSC method of power allocation.

Both the two approaches of OSC and Adaptive have a significant improvement in the BER by having a higher priority to notify closer vehicles than distant ones. Therefore we observed in section 4.2.4 that not using a network congestion control algorithm have higher reception rate but the negative impact of this is shown here by having a higher beacon error rate.

The improvement on the network congestion control algorithms, more specifically the adaptive method have regarding improving the beacon error rate is both approaches choose to prioritize closer traffic by reducing the transmission power. The adaptive approach further improves on this by introducing the cycle of power transmission along with having a speed factor, in return, we would have a gradual increase of communication which reduces the BER.

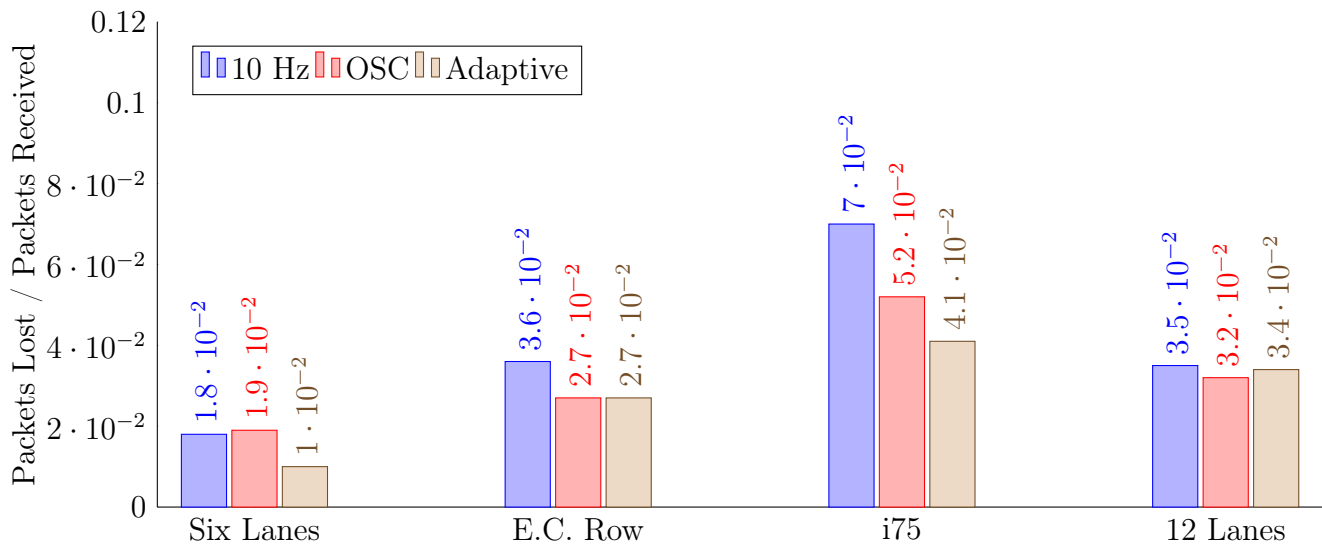


Figure 4.8: Beacon Error Rate

## 4.2.6 Channel Busy Time

When a vehicle wants to transmit a message, it senses the channel and waits for a clear response for it to send. This is managed by the carrier sense threshold algorithm CSMA. According to the CSMA, as shown in Appendix D, When the vehicle encounters a busy response from the channel, it waits for a random time known as *back-off*, then it tries again.

Channel busy time indicates the amount of time in seconds vehicles encounter a busy response from the channel. Network congestion control algorithms aim to keep this number as low as possible. We would expect that not using a control algorithm would have a higher busy time, and this is further supported in Figure 4.9. The MAC module in VEINS 4.6 records scalar value at the end of the simulation known as *totalBusyTime*, divided by the total simulation time would indicate the amount of time the MAC was treated as busy.

The effect of not attempting to reduce channel congestion is further illustrated in Figure 4.8 by having a significant higher busy response from the channel sense threshold. Therefore, it is important to examine the CBR ratio when evaluating the network congestion of inter-vehicular communication.

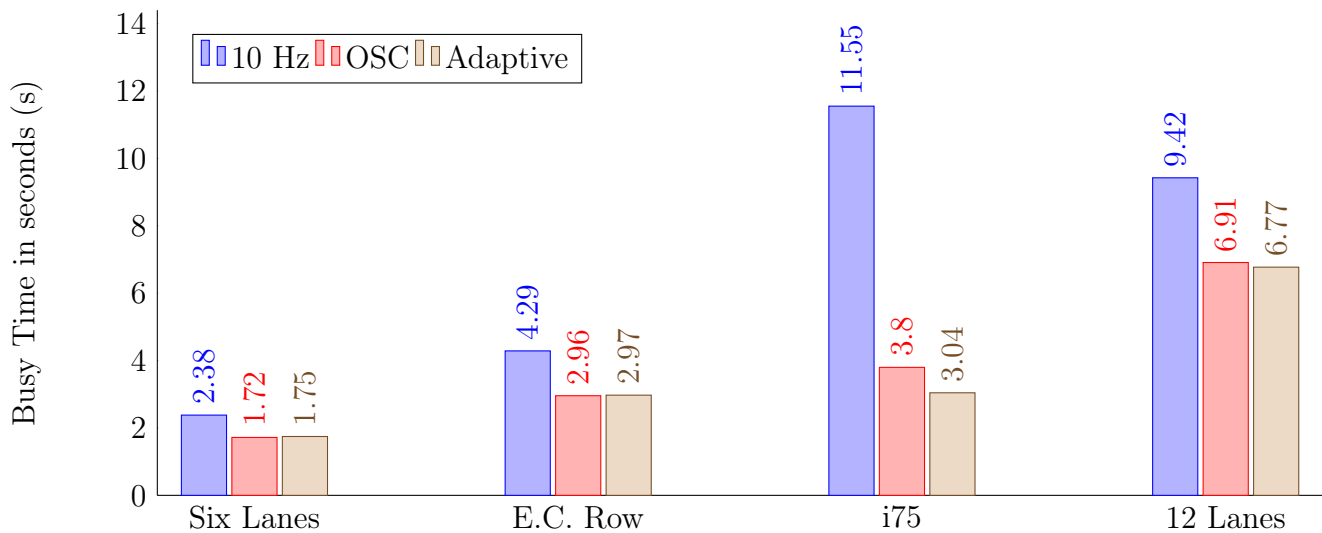


Figure 4.9: Channel Busy Ratio (CBR)



### 4.2.7 Inter-Packet Delay (IPD)

Our goal by introducing a cycle fashion approach of increasing transmission power and adapting it to the vehicle speed is to reduce the inter-packet delay experienced by using the OSC power adaptation algorithm. We believe by using this approach; it would help to reduce the rapid increase of IPD in order to observe a gradual increase in the IPD value as distance increases. However, following this approach cannot be done in the expenses of sacrificing other parameters such as BRR and BER. We aim to maintain other parameters when compared to OSC while improving on the IPD. The gradual increase in IPD in the Adaptive algorithm and the sudden increase of IPD in OSC algorithm are done intentionally in the essence to sacrifice awareness for distance vehicles by reducing the travel distance of packets to reduce channel load.

We calculate the inter-packet delay by recording the distance packets travel divide them with an increment of 20 meters. For example, we record the number of total packets travelled a distance of 0 - 20 meters, 20 - 40 meters, 40 - 60 meters and so on.

After simulating all four scenarios, our hypothesis of having a gradual increase of IPD over a sudden rise in IPD we experience in OSC was confirmed by obtained from the simulation environments and are further illustrated in Figures 4.10, 4.11 4.12 and 4.13.

OSC Algorithm improves the channel congestion compared to not using a network congestion control but this happens with sacrificing awareness for distant vehicles. OSC Suffer from a rapid increase in the IPD numbers for distant vehicles as shown in Figures 4.10, 4.11 4.12 and 4.13. This can pose a serious issue in the safety application of vehicular networks. Therefore, our work has shown so far to maintain the channel usage ratio and beacon error rate compared to OSC but further improved on the IPD values by having a gradual increase rather than a rapid increase.

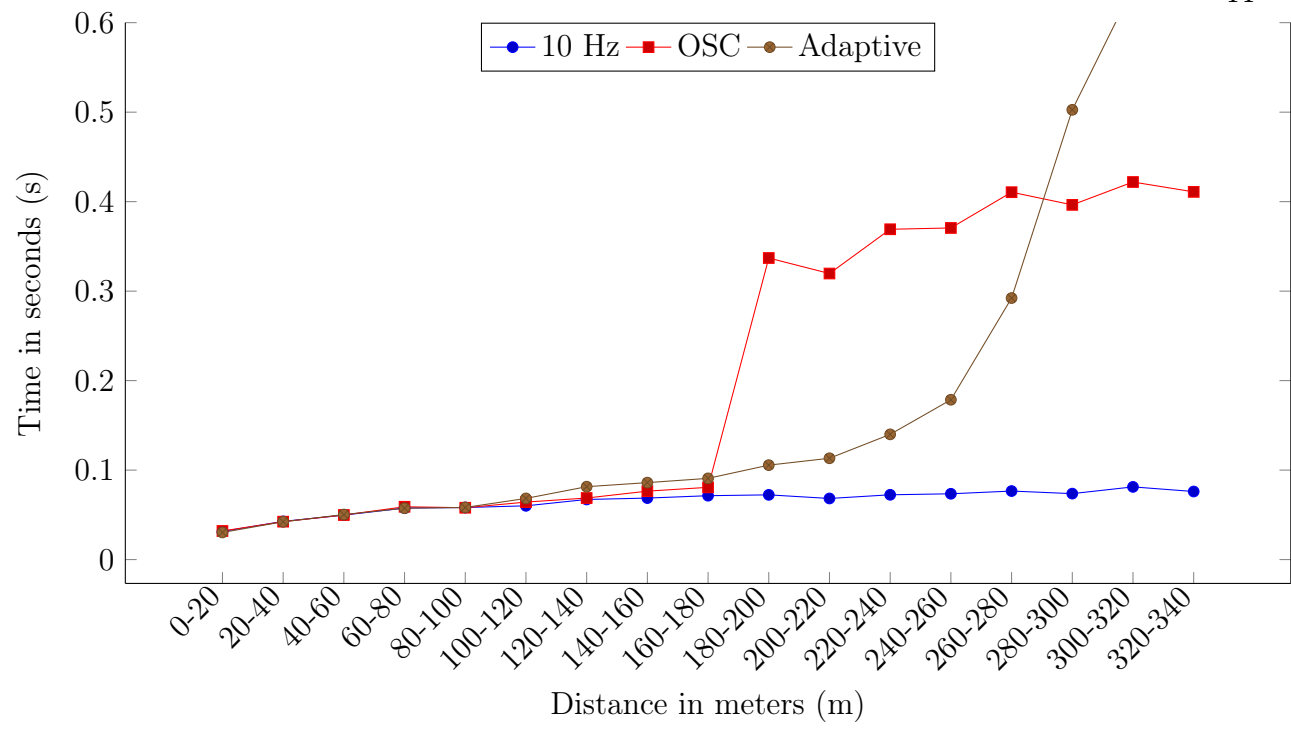


Figure 4.10: Six Lane Inter-Packet Delay

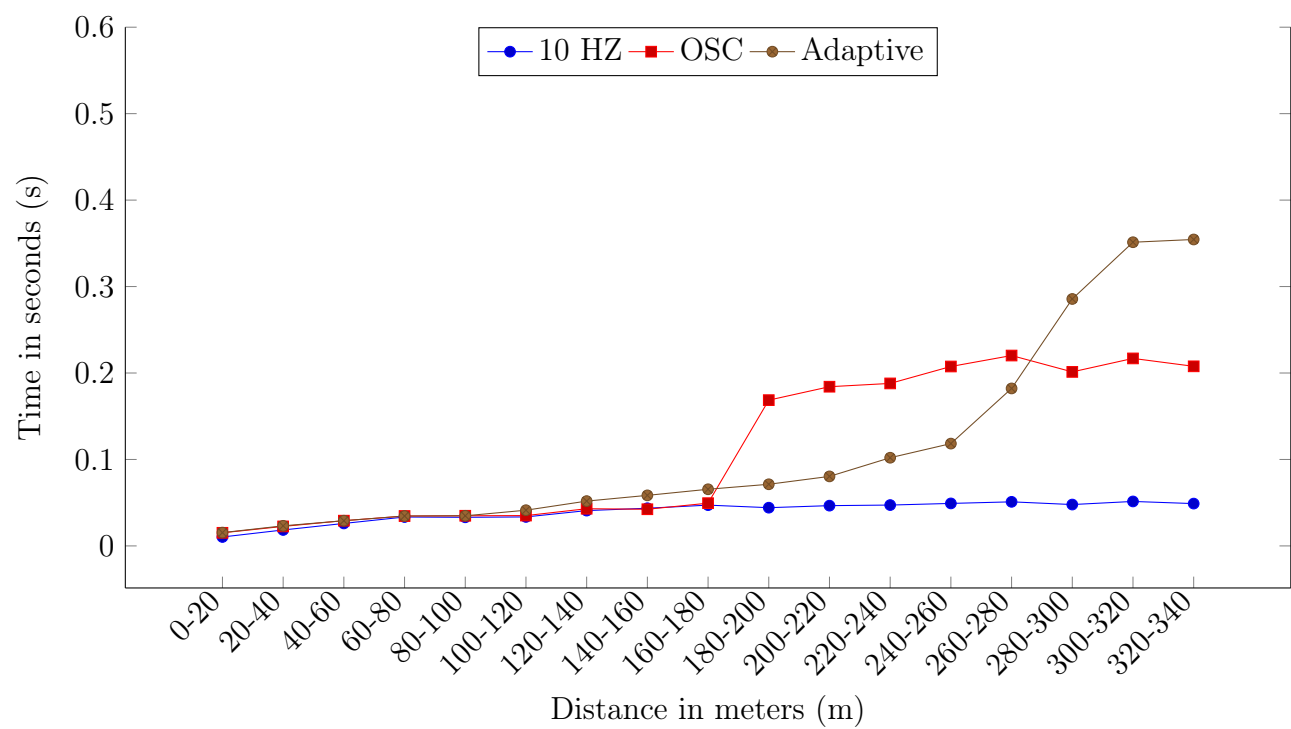


Figure 4.11: 12 Lane Inter-Packet Delay

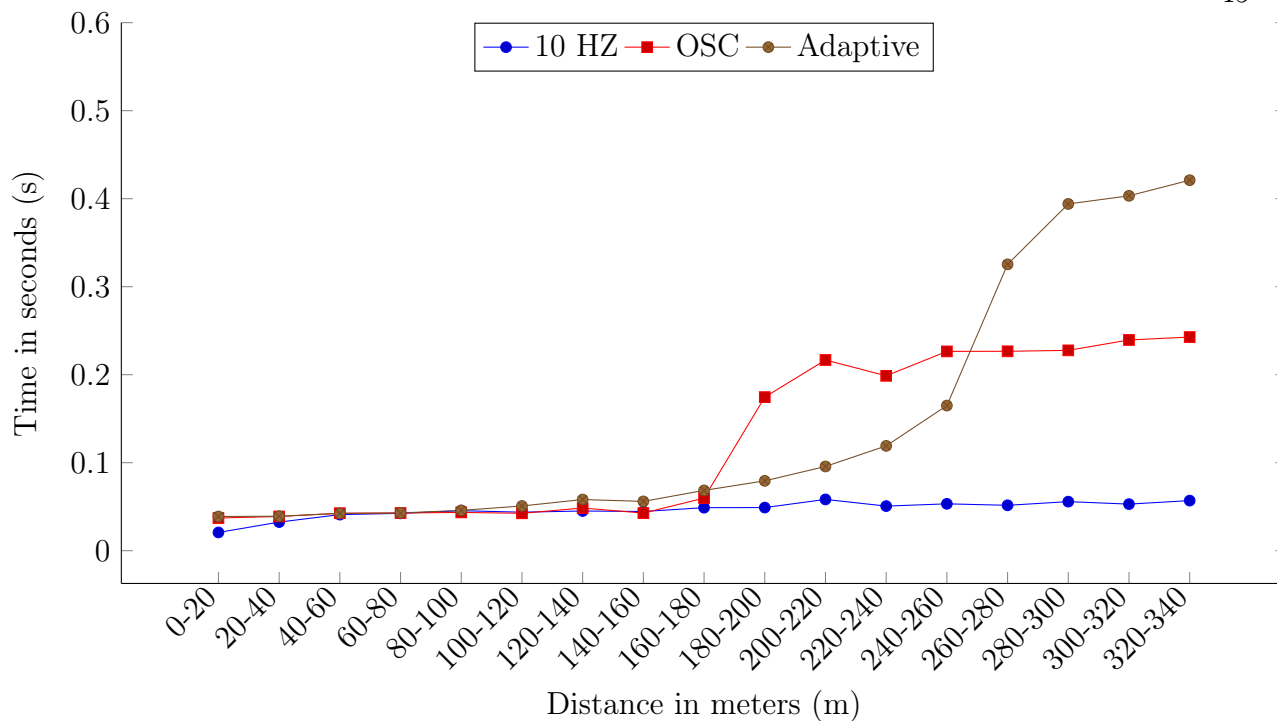


Figure 4.12: E.C. Row Inter-Packet Delay

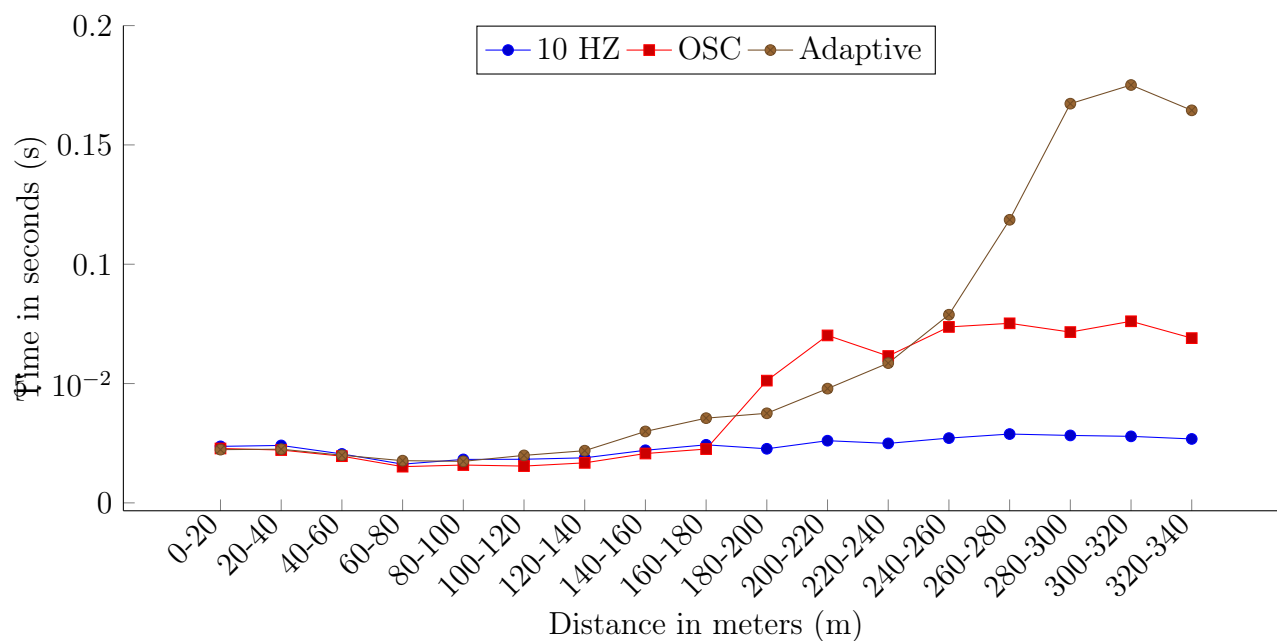


Figure 4.13: Interstate i-75 Inter-Packet Delay

## 4.2.8 Fairness

Algorithm Fairness was calculated using Jain's fairness index where the fairness of 1 means the algorithm is fair. By examining Figure 4.14 we can conclude that both

algorithms perform similarly regarding fairness. Fairness means that everytime a new vehicle is created and introduced to the network, it has a fair share of resources of the network and can broadcast messages. Please refer to Figure E.4 for fairness results.

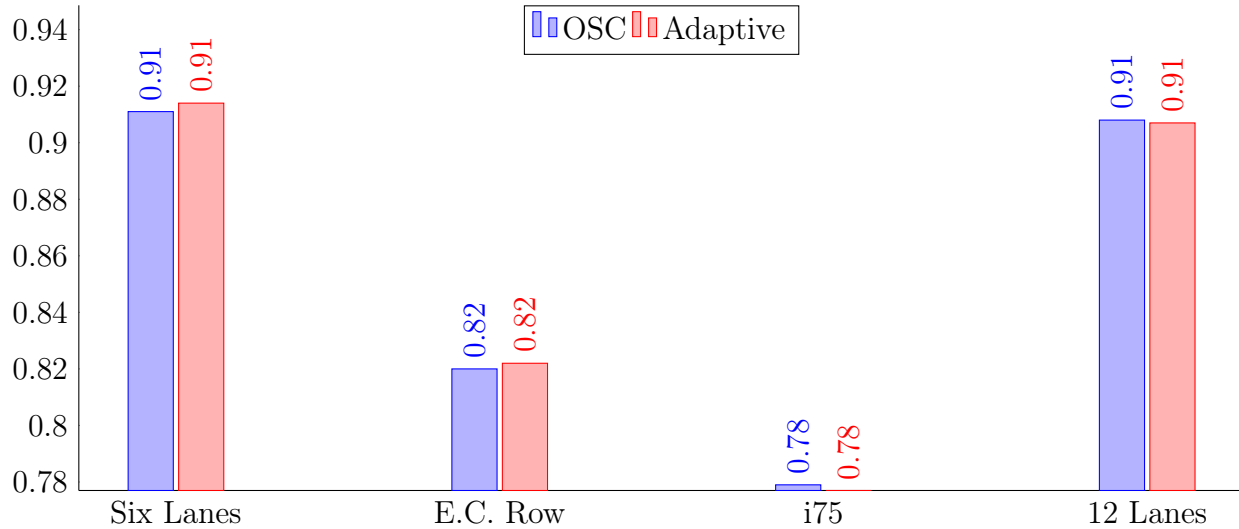


Figure 4.14: Algorithm Fairness

Fairness is calculated by measuring each vehicle's throughput during the simulation time. Jain's fairness was used to compute the fairness for both algorithms, the equation of Jain's index is shown in Figure 4.15. Where  $n$  is the number of vehicles observed,  $\mathcal{X}_i$  is a vehicle's throughput.

$$\frac{\left(\sum_{i=1}^n \mathcal{X}_i\right)^2}{n \times \sum_{i=1}^n \mathcal{X}_i^2}$$

Figure 4.15: Jain's Fairness Index

# Chapter 5

## Conclusion and Future Work

### 5.1 Conclusion

The intentions driven by the Adaptive power algorithm to modify power transmission was to reduce channel congestion in the overall picture. It also improved on the Oscillating power model to reduce the inter-packet delay surge when it switches from lower power to high power packets. The oscillating power model sacrifices awareness from distant vehicles while maintaining awareness of closer ones. While this approach showed a significant improvement on channel congestion when comparing to not using a congestion control algorithm but from what we saw in Chapter 4, this works on the expense of sacrificing awareness for distant vehicles. Our results demonstrate improvement in channel utilization; hence improvement to the safety application because a) We were able to reduce packets lost, b) Reduce channel busy time, and c) Maintain the inter-packet delay for closer vehicles and gradually increase the IPD for distant vehicles.

Our goal was to maintain the channel usage OSC had accomplished and improved on the inter-packet delay. The sudden increase of IPD the OSC suffers from for mid-range communication can render an unreliable vehicular communication due to the sudden drop in awareness. Additionally, our goal was to also sacrifice awareness for distant vehicles but

at a gradual rate to eliminate the sudden increase of IPD. Our algorithm showed that it is able to maintain a low channel congestion and reduce the sudden increase of IPD by having a gradual increase as the distance gets farther. Further, Adaptive was able to reduce the packet error rate and channel busy time when compared to the 10 Hz, and maintained similar packet error rate, Channel utilization and fairness when compared to OSC.

## 5.2 Future Work

Various different approaches should be considered to handle channel load in DSRC. In this paper, we examined two different approaches congestion and another without mitigating channel congestion. In Chapter four, we outlined the results from simulation, which showed each approach has its advantages and disadvantages; as a result, there is no *ideal* approach, hence there are other approaches that would help to reduce channel congestion.

To give an example, our algorithm works on adjusting the transmission power level, another approach could be tested is to use a similar idea but involving adjusting the transmission rate and having a constant transmission power. Further, both approaches to mitigate transmission power and transmission rate can be explored in an urban scenario where streets in such scenario would intersect. Both of the approaches mentioned above can be further improved by adapting vehicle density in a specified short range

# Appendix A

## OMNeT Configuration

---

```
[General]

cmdenv-express-mode = true

cmdenv-autoflush = true

cmdenv-status-frequency = 1s

ned-path = .

network = RSUEExampleScenario

#####
#           Simulation parameters           #
#####

debug-on-errors = true

print-undisposed = false

sim-time-limit = 100s

**.scalar-recording = true

**.vector-recording = true

**.debug = false

**.coreDebug = false

*.playgroundSizeX = 9000m

*.playgroundSizeY = 9000m
```

```

*.playgroundSizeZ = 50m
#####
# Annotation parameters                                     #
#####
*.annotations.draw = false
#####
#           TraCIScenarioManager parameters                 #
#####
*.manager.updateInterval = 1s
*.manager.host = "localhost"
*.manager.port = 9999
*.manager.autoShutdown = true
*.manager.margin = 25
*.manager.launchConfig = xmldoc("sumo-launchd.launch.xml")
#####
#           11p specific parameters                         #
#                                                         #
#           NIC-Settings                                    #
#####
*.connectionManager.sendDirect = true
*.connectionManager.maxInterfDist = 2600m
*.connectionManager.drawMaxIntfDist = false
*.*.nic.mac1609_4.useServiceChannel = false
*.*.nic.mac1609_4.txPower = 10mW
*.*.nic.mac1609_4.bitrate = 6Mbps
*.*.nic.phy80211p.sensitivity = -89dBm
*.*.nic.phy80211p.useThermalNoise = true
*.*.nic.phy80211p.thermalNoise = -110dBm
*.*.nic.phy80211p.decider = xmldoc("../config.xml")

```



```

***.nic.phy80211p.analogueModels = xmldoc("../config.xml")
***.nic.phy80211p.usePropagationDelay = true
#####
#                               WaveAppLayer                               #
#####
*.node[*].applType = "TraCIDemo11p"
*.node[*].applType = "MyVeinsApp"
*.node[*].appl.headerLength = 80 bit
*.node[*].appl.debug = true
*.node[*].appl.sendBeacons = true
#*.node[*].appl.sendData = true
*.node[*].appl.dataOnSch = false
*.node[*].appl.beaconInterval = 0.1s
#####
#                               Mobility                               #
#####
*.node[*].veinsmobilityType.debug = true
*.node[*].veinsmobility.x = 0
*.node[*].veinsmobility.y = 0
*.node[*].veinsmobility.z = 1.895
*.node[*0].veinsmobility.accidentCount = 1
*.node[*0].veinsmobility.accidentStart = 75s
*.node[*0].veinsmobility.accidentDuration = 50s
*.node[*].appl.dataOnSch = true

[Config Adaptive]
[Config OSC]
*.node[*].appl.algo = 1

```

---

## Appendix B

# SUMO Configuration

---

```

<routes>
  <vType accel="10.0" decel="5.0" id="Car" length="2.0" maxSpeed="100.0" sigma="
    0.7" />
  <route id="easttraffic" edges="east2 east1" />
  <route id="westtraffic" edges="west1 west2" />
  <flow id="type1" color="blue" begin="0" end="400" period="0.01" type="Car"
    departLane="random" from="west1" to="west2" />
  <flow id="type2" color="1,1,0" begin="0" end="400" period="0.01" type="Car"
    departLane="random" from="east2" to="east1" />
</routes>

```

---

Figure B.1: SUMO Route Configuration

---

```

<configuration>
  <input>
    <net-file value="sim1.net.xml"/>
    <route-files value="sim1.rou.xml"/>
  </input>
  <time>
    <begin value="0"/>
    <end value="1000"/>
  </time>
  <gui_only>
    <start value="false"/>
  </gui_only>
</configuration>

```

---

Figure B.2: SUMOCFG

# Appendix C

## Algorithm Source File

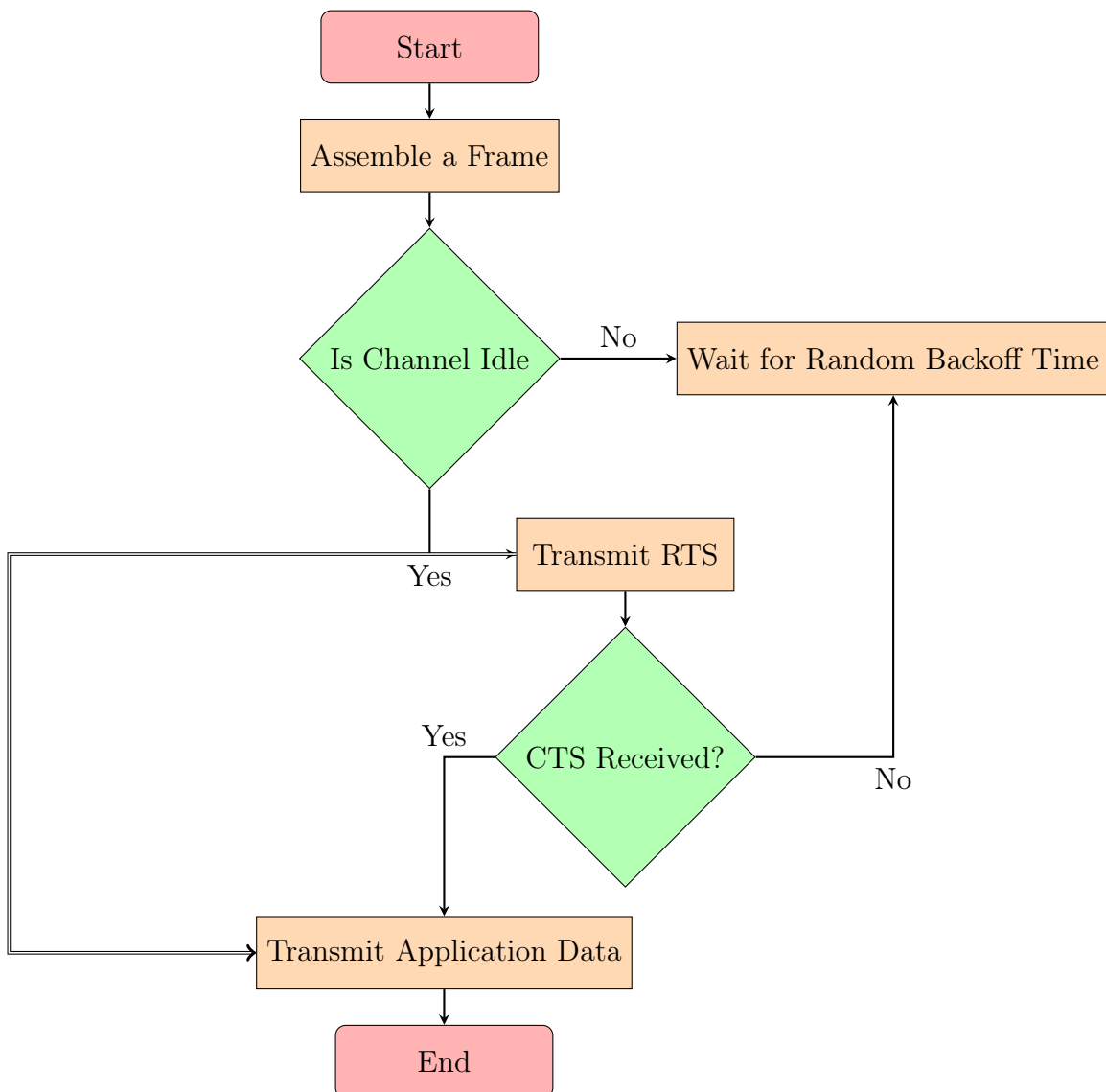
---

```
void MyVeinsApp::AdaptiveTX()
    float speedFactor = 0;
    double vehicleSpeed = mobility->getSpeed();
    if (vehicleSpeed > 90)
        speedFactor = 1.4;
    else if(vehicleSpeed > 60)
        if (vehicleSpeed <= 90)
            speedFactor = 1.2;
    else if (vehicleSpeed > 40)
        if (vehicleSpeed <= 60)
            speedFactor = 1.1;
    else if(vehicleSpeed <= 40)
        speedFactor = 1.05;
    if (local_tx == 7)
        mac->setTxPower(10);
        local_tx = 1;
    else
        mac->setTxPower(local_tx*speedFactor);
        local_tx ++;
```

---

## Appendix D

### CSMA Flowchart



# Appendix E

## Scalar Results

	Six Lanes	12 Lanes	Edward Charles Row	Interstate 75
Sent Packets	62680	125130	84130	130310
Received Packets	2214417	8776991	3965083	10594877
Lost Packets	40369	303907	143017	746523
Busy Time	239.15	942.47	428.71	1155.15
Packet Receive Rate	35.33	70.14	47.13	81.31
Packet Error Rate	0.018	0.035	0.036	0.070

Table E.1: 10 Hz Scalar Results

	Six Lanes	12 Lanes	Edward Charles Row	Interstate 75
Sent Packets	62680	125130	84130	130310
Received Packets	1573524	6277327	2717921	7325601
Lost Packets	29755	198309	72275	379863
Busy Time	171.88	677.28	295.65	796.89
Packet Receive Rate	25.10	50.17	32.31	56.22
Packet Error Rate	0.019	0.032	0.027	0.052

Table E.2: OSC Scalar Results

	Six Lanes	12 Lanes	Edward Charles Row	Interstate 75
Sent Packets	62680	125130	84130	130310
Received Packets	1606925	6402773	2731476	7444846
Lost Packets	16421	215572	75023	304342
Busy Time	174.65	691.01	297.34	805.72
Packet Receive Rate	25.64	51.17	32.47	57.13
Packet Error Rate	0.010	0.034	0.027	0.041

Table E.3: Adaptive Scalar Results

	OSC	Adapt
Six Lanes	0.911	0.914
E.C. Row	0.820	0.822
I75	0.779	0.777
12 Lanes	0.908	0.907

Table E.4: Fairness

# Bibliography

- [1] Gaurav Bansal, John B Kenney, and Charles E Rohrs. Limeric: A linear adaptive message rate algorithm for dsrc congestion control. *IEEE Transactions on Vehicular Technology*, 62(9):4182–4197, 2013.
- [2] Gaurav Bansal, Hongsheng Lu, John B Kenney, and Christian Poellabauer. Embarc: Error model based adaptive rate control for vehicle-to-vehicle communications. In *Proceeding of the tenth ACM international workshop on Vehicular inter-networking, systems, and applications*, pages 41–50. ACM, 2013.
- [3] Michael Behrisch, Laura Bieker, Jakob Erdmann, and Daniel Krajzewicz. Sumo—simulation of urban mobility: an overview. In *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind, 2011.
- [4] C Belagal Math, H Li, SM Heemstra, and IGMM Niemegeers. V2x application-reliability analysis of data-rate and message-rate congestion control algorithms. 2017.
- [5] Pedro Fernandes and Urbano Nunes. Vehicle communications: A short survey. *IADIS Telecommunications, Networks and Systems, Lisbon*, pages 134–138, 2007.
- [6] Mordechai Haklay and Patrick Weber. Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.

- [7] Jianhua He, Hsiao-Hwa Chen, Thomas M Chen, and Wenqing Cheng. Adaptive congestion control for dsrc vehicle networks. *IEEE communications letters*, 14(2):127–129, 2010.
- [8] John B Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, 2011.
- [9] John B Kenney, Gaurav Bansal, and Charles E Rohrs. Limeric: a linear message rate control algorithm for vehicular dsrc systems. In *Proceedings of the Eighth ACM international workshop on Vehicular inter-networking*, pages 21–30. ACM, 2011.
- [10] Bernhard Kloiber, Jérôme Härri, Thomas Strang, Stephan Sand, and Cristina Rico Garcia. Random transmit power control for dsrc and its application to cooperative safety. *IEEE Transactions on Dependable and Secure Computing*, 13(1):18–31, 2016.
- [11] Timo Kosch, Christian J Adler, Stephan Eichler, Christoph Schroth, and Markus Strassberger. The scalability problem of vehicular ad hoc networks and how to solve it. *IEEE Wireless Communications*, 13(5), 2006.
- [12] Chetan Belagal Math, Hong Li, Sonia Heemstra de Groot, and Ignas Niemegeers. Fair decentralized data-rate congestion control for v2v communications. In *Telecommunications (ICT), 2017 24th International Conference on*, pages 1–7. IEEE, 2017.
- [13] matthew.lynberg.ctr@dot.gov. Crash injury research, Apr 2017.
- [14] Jungme Park, Zhihang Chen, Leonidas Kiliaris, Ming L Kuang, M Abul Masrur, Anthony M Phillips, and Yi Lu Murphey. Intelligent vehicle power control based on machine learning of optimal control parameters and prediction of road type and traffic congestion. *IEEE Transactions on Vehicular Technology*, 58(9):4741–4756, 2009.



- [15] José Santa, Antonio F Gómez-Skarmeta, and Marc Sánchez-Artigas. Architecture and evaluation of a unified v2v and v2i communication system based on cellular networks. *Computer Communications*, 31(12):2850–2861, 2008.
- [16] Christoph Sommer, Reinhard German, and Falko Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, January 2011.
- [17] Heecheol Song and Hwang Soo Lee. A survey on how to solve a decentralized congestion control problem for periodic beacon broadcast in vehicular safety communications. In *Advanced Communication Technology (ICACT), 2013 15th International Conference on*, pages 649–654. IEEE, 2013.
- [18] Sundar Subramanian, Marc Werner, Shihuan Liu, Jubin Jose, Radu Lupoai, and Xinzhou Wu. Congestion control for vehicular safety: synchronous and asynchronous mac algorithms. In *Proceedings of the ninth ACM international workshop on Vehicular inter-networking, systems, and applications*, pages 63–72. ACM, 2012.
- [19] Antony Tang and Alice Yip. Collision avoidance timing analysis of dsrc-based vehicles. *Accident Analysis & Prevention*, 42(1):182–195, 2010.
- [20] Tessa Tielert, Daniel Jiang, Hannes Hartenstein, and Luca Delgrossi. Joint power/rate congestion control optimizing packet reception in vehicle safety communications. In *Proceeding of the tenth ACM international workshop on Vehicular inter-networking, systems, and applications*, pages 51–60. ACM, 2013.
- [21] András Varga and Rudolf Hornig. An overview of the omnet++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 60. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

- [22] Anna Maria Vegni and Thomas DC Little. Hybrid vehicular communications based on v2v-v2i protocol switching. *International Journal of Vehicle Information and Communication Systems*, 2(3-4):213–231, 2011.
- [23] Jordan Tyler Willis. Decentralized congestion control algorithm for vehicle to vehicle networks using oscillating transmission power. 2015.
- [24] Xue Yang, L Liu, Nitin H Vaidya, and Feng Zhao. A vehicle-to-vehicle communication protocol for cooperative collision warning. In *Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on*, pages 114–123. IEEE, 2004.
- [25] Jing Zhu and Sumit Roy. Mac for dedicated short range communications in intelligent transport system. *IEEE Communications Magazine*, 41(12):60–67, 2003.

# Vita Auctoris

Maan grew up in Windsor, Ontario Canada. He attended St. Joseph's Catholic High School. He later pursued a B.Sc in Biology and Chemistry followed by a B.Sc. in Computer Science with Software Engineering at the University of Windsor. His passion for computer science and working on vehicles as a hobby pushed him to pursue a Master's degree in Computer Science.