Improving TCP Behaviour to Non-invasively Share Spectrum with Safety Messages in VANET

M. Shahid Anwer

School of Systems Engineering
University of Reading
United Kingdom

This Thesis is Submitted for the Degree of
Doctor of Philosophy

May-2017

Declaration

I confirm that this dissertation is my own work and the use of all material from other sources has been properly and fully acknowledged.

M. Shahid Anwer

Dedication

To my loving parents for their ever-present blessings,

and

To my beloved wife for her unconditional love.

Acknowledgements

First and foremost, I shall remain ever grateful to the mercy and compassion God Almighty has shown, and to Him who has graciously bestowed upon me immense strength and indomitable courage to carry through this Ph.D. project against all odds.

I offer my sincere thanks to Professor Chris Guy for his invaluable guidance, advice and motivation right from the start of research. His extensive research experience and his benevolent support have certainly helped me reach up to this level.

My thanks are also due to Professor Simon Sherratt and Dr. Sillas Hadjiloucas for encouraging and enabling me to finally arrive at the goal post, which seemed to be shifting strangely at times, especially after the departure of Professor Chris Guy from the University.

Last but certainly not least, I wish to place on record the great sacrifice made by my wife, who tirelessly and single-mindedly worked to finance my rather long, tedious and expensive Ph.D. journey. Torn between work and home, she toiled patiently and courageously for long, to see that her husband eventually emerges victorious in his research endeavours. Without this selfless service, support and encouragement on her part, this thesis simply would not have seen the light of day.

Abstract

There is a broad range of technologies available for wireless communications for moving vehicles, such as Worldwide Interoperability for Microwave Access (WiMax), 3G, Dedicated Short Range Communication (DSRC)/ Wireless Access for Vehicular Environment (WAVE) and Mobile Broadband Wireless Access (MBWA). These technologies are needed to support delay-sensitive safety related applications such as collision avoidance and emergency breaking. Among them, the IEEE802.11p standard (aka DSRC/WAVE), a Wi-Fi based medium RF range technology, is considered to be one of the best suited draft architectures for time-sensitive safety applications.

In addition to safety applications, however, services of non-safety nature like electronic toll tax collection, infotainment and traffic control are also becoming important these days. To support delay-insensitive infotainment applications, the DSRC protocol suite also provides facilities to use Internet Protocols. The DSRC architecture actually consists of WAVE Short Messaging Protocol (WSMP) specifically formulated for real-time safety applications as well as the conventional transport layer protocols TCP/UDP for non-safety purposes. But the layer four protocol TCP was originally designed for reliable data delivery only over wired networks, and so the performance quality was not guaranteed for the wireless medium, especially in the highly unstable network topology engendered by fast moving vehicles. The vehicular wireless medium is inherently unreliable because of intermittent disconnections caused by moving vehicles, and in addition, it suffers from multi-path and fading phenomena (and a host of others) that greatly degrade the network performance.

One of the TCP problems in the context of vehicular wireless network is that it interprets transmission errors as symptomatic of an incipient congestion situation and as a result, reduces the throughput deliberately by frequently invoking slow-start congestion control algorithms. Despite the availability of many congestion control mechanisms to address this problem, the conventional TCP continues to suffer from poor performance when deployed in the Vehicular Ad-hoc Network (VANET) environment.

Moreover, the way non-safety applications, when pressed into service, will treat the existing delay-sensitive safety messaging applications and the way these two types of applications interact between them are not (well) understood, and therefore, in order for them to coexist, the implication and repercussion need to be examined closely. This is especially important as IEEE 802.11p standards are not designed keeping in view the issues TCP raises in relation to safety messages.

This dissertation addresses the issues arising out of this situation and in particular confronts the congestion challenges thrown up in the context of heterogenous communication in VANET environment by proposing an innovative solution with two optimized congestion control algorithms. Extensive simulation studies conducted by the author shows that both these algorithms have improved TCP performance in terms of metrics like Packet Delivery Fraction (PDF), Packet Loss and End-to-End Delay (E2ED), and at the same time they encourage the non-safety TCP application to behave unobtrusively and cooperatively to a large extent with DSRC's safety applications.

The first algorithm, called vScalable-TCP – a modification of the existing TCP-Scalable variant – introduces a reliable transport protocol suitable for DSRC. In the proposed approach, whenever packets are discarded excessively due to congestion, the slow-start mechanism is purposely suppressed temporarily to avoid further congestion and packet loss. The crucial idea here is how to adjust and regulate the behaviour of vScalable-TCP in a way that the existing safety message flows are least disturbed. The simulation results confirm that the new vScalable-TCP provides better performance for real-time safety applications than TCP-Reno and other TCP variants considered in this thesis in terms of standard performance metrics.

The second algorithm, named vLP-TCP – a modification of the existing TCP-LP variant – is designed to test and demonstrate that the strategy developed for vScalable-TCP is also compatible with another congestion control mechanism and achieves the same purpose. This expectation is borne out well by the simulation results. The same slow-

start congestion management strategy has been employed but with only a few amendments. This modified algorithm also improves substantially the performance of basic safety management applications.

The present work thus clearly confirms that both vScalable-TCP and vLP-TCP algorithms – the prefix ‘v’ to the names standing for ‘vehicular’ – outperform the existing unadorned TCP-Scalable and TCP-LP algorithms, in terms of standard performance metrics, while at the same time behaving in a friendly manner, by way of sharing bandwidth non-intrusively with DSRC safety applications. This paves the way for the smooth and harmonious coexistence of these two broad, clearly incompatible or complementary categories of applications – *viz.* time-sensitive safety applications and delay-tolerant infotainment applications – by narrowing down their apparent *impedance* or *behavioural mismatch*, when they are coerced to go hand in hand in a DSRC environment.

Table of Contents

Declaration i
Dedication ii
Acknowledgements.. iii
Abstract iv
List of Tables.. x
List of Figures... xi
List of Abbreviations xiii

Chapter 1
Introduction 1
1.1 An Overview of DSRC 1
1.2 DSRC Safety Messaging Services... 3
1.3 DSRC Non-safety Application in VANET 4
1.4 Research Motivation 5
1.5 Challenges of Deploying TCP in VANET... 6
1.6 Research Contribution 9
1.7 Dissertation Organization 11

Chapter 2
Background Study and Related Work 13
2.1 MANET 13
2.2 VANET 15
2.3 The DSRC for VANET.. 19
2.4 Architecture of DSRC.. 21
  2.4.1 PHY Layer 21
  2.4.2 MAC layer.. 23
  2.4.3 IEEE P1609.x.. 25
  2.4.4 Networking Services... 25
  2.4.5 Authentication and Association Process in DSRC 26
2.5 Strength and Weakness of DSRC.. 28
2.6 Overview of TCP... 29
  2.6.1 TCP’s Congestion Control Mechanisms..................................... 30
2.7 TCP Variants 32
  2.7.1 TCP-Reno.. 32
  2.7.2 TCP-BIC 33
  2.7.3 TCP-Scalable... 34
  2.7.4 TCP-Westwood 35
  2.7.5 TCP-HighSpeed... 35

2.8 TCP in VANET 37
2.9 Related Work.. 39
2.9.1 Related Work on Message Interval Rate and Modulation Scheme over VANET………………………………………………………………..39
2.9.2 Related Work on Use of TCP over VANET....................................... 43
2.10 Chapter Summary 50

Chapter 3.

VANET Simulation, Performance Metrics and Experimental Scenarios........... 53
3.1 Traffic Environment for VANET simulation.. 53
3.2 Vehicular Mobility Generator 55
3.3 MobiSim Mobility Simulator for VANET.. 55
3.4 Freeway Mobility Model Using MobiSim... 57
3.5 IEEE 802.11p Based VANET Simulation using NS2 58
3.5.1 DSRC's Modulation Scheme in NS2 60
3.5.2 Data Traffic Source for Safety Application 61
3.5.3 DSRCs Multi-channel Support in NS2... 62
3.5.4 NAM Output in NS2... 63
3.6 Routing Protocol in Freeway Scenario 64
3.7 Propagation Model for Freeway Scenario 67
3.8 Performance Evaluation Metrics 68
3.9 Simulation Scenarios and Experimental Parameters 70
3.9.1 Simulation Scenario-1... 70
3.9.2 Simulation Scenario-2... 71
3.10 Chapter Summary 72

Chapter 4

Performance Analysis of BSM and Non-safety Applications.............................. 74
4.1 A Study on the Performance of DSRC Safety Messaging Application........ 74
4.1.1 Introduction... 75
4.1.2 One Hop Broadcast for BSM 76
4.1.3 Simulation Scenario... 77
4.1.4 Simulation Results and Discussion 78
4.1.4.1 Performance with respect to Modulation Schemes and with Number of Nodes... 78
4.1.4.2 Performance with respect to Message Intervals and with Number of Nodes…………………………………………………………..81
4.1.5 Section Summary... 83
4.2 A Study on the Performance of TCP based Application over DSRC based VANET 85
4.2.1 Introduction... 85

4.2.2 Simulation Scenario 86
4.2.3 Simulation Results and Discussion 86
4.2.4 Section Summary 96
4.3 A Study on the Impact of TCP Flow over Periodic BSM 97
4.3.1 Introduction 97
4.3.2 Simulation Scenario 98
4.3.3 Simulation Results and Discussion 99
4.3.4 Section Summary 104
4.4 Chapter Summary 104

Chapter 5
vScalable-TCP: An Optimized TCP-Scalable Algorithm for Sharing Bandwidth Cooperatively with Safety Applications 107
5.1 Introduction 107
5.2 Problem Description 109
5.3 Analysis & Implementation of TCP-Scalable on NS2 Linux TCP 111
5.4 Design & Implementation of Optimized vScalable-TCP Algorithm 116
5.5 The Performance of Non-safety Application over vScalable-TCP 119
5.6 A Study on the Impact of vScalable-TCP on Periodic BSM 125
5.7 Chapter Summary 130

Chapter 6
vLP-TCP: An Optimized Low Priority TCP Algorithm to Share Bandwidth Cooperatively with Safety Application 132
6.1 Introduction 132
6.2 Analysis & Implementation of TCP-LP on NS2 Linux TCP 134
6.3 Design & Implementation of Optimized vLP-TCP Algorithm 138
6.4 The Performance of Non-safety Messaging Application over vLP-TCP ... 140
6.5 A Study on the Impact of vLP-TCP on Periodic BSM 143
6.6 Chapter Summary 147

Chapter 7
Conclusion and Future Work 148
7.1 Introduction 148
7.2 Summary of Results and Contributions 149
7.3 Direction for Future Work 151

List of Tables

Table 2.1: Key characteristics of MANET and VANET 16
Table 2.2: Comparative study of wireless technology suitable for VANET[17] 20
Table 2.3: Comparison of PHY layer parameters of IEEE802.11p and IEEE802.11a 23
Table 2.4: 802.11p and 802.11a connection duration [66] 27
Table 2.5: Strength and weakness of DSRC 28
Table 3.1: Modulation index used in NS2.35 for 802.11p simulation 61
Table 3.2: Scenario-1 simulation parameters for RSU & vehicles 71
Table 3.3: Scenario-2 simulation parameters for RSU & vehicles 72
Table 5.1: Variable that control behaviour of TCP-Scalable 114

List of Figures

Figure 1.1: The distinction between WAVE & DSRC as per OSI model 2
Figure 1.2: Safety and Non-Safety (NS) channel usage scenario 8
Figure 2.1: An example of MANET and Infrastructure wireless network 15
Figure 2.2: V2V, V2I and V2X Communication [49] 18
Figure 2.3: Architecture of DSRC 21
Figure 2.4: DSRC/WAVE Operating Channels 22
Figure 3.1: MobiSim pluggable architecture 57
Figure 3.2: MobiSim showing a 8-lane freeway traffic scenario 58
Figure 3.3: NS2 and MobiSim interaction process 60
Figure 3.4: The packet delivery rate [148] 63
Figure 3.5: NAM showing 50 vehicles on a bidirectional freeway scenario 64
Figure 3.6: AODV Route discovery mechanism 66
Figure 4.1: Sent packets in different modulation and No. of vehicles 78
Figure 4.2: Received packets in different modulation and No. of vehicles 78
Figure 4.3: E2ED in different modulation and No. of vehicles 78
Figure 4.4: PDF in different modulation and No. of vehicles 78
Figure 4.5: Average Throughput in different modulation and No. of vehicles 79
Figure 4.6: Sent packets at BSM intervals and No. of vehicles 81
Figure 4.7: Received packets at BSM intervals and No. of vehicles 81
Figure 4.8: E2ED at BSM intervals and No. of vehicles 81
Figure 4.9: PDF at BSM intervals and No. of vehicles 81
Figure 4.10: Average Throughput at BSM intervals and No. of vehicles 82
Figure 4.11: cwnd dynamic of TCP variants 87
Figure 4.12: RTT dynamics of TCP variants 89
Figure 4.13: Sent Packets of TCP variants at BSM intervals 90
Figure 4.14: Received Packets of TCP variants at BSM intervals 90
Figure 4.15: E2ED of TCP variants at BSM intervals 91
Figure 4.16: Average Throughput of TCP variants at BSM intervals 91
Figure 4.17: Total dropped packet at different BSM intervals 95
Figure 4.18: PDF of TCP variants at different BSM intervals 95
Figure 4.19: BSM Sent Packets in the presence of TCP variants 99
Figure 4.20: BSM Received Packets in the presence of TCP variants 99
Figure 4.21: E2ED of BSM in the presence of TCP variants 99
Figure 4.22: Average Throughput of BSM in the presence of TCP variants 99
Figure 4.23: PDF of BSM in the presence of TCP variants 103
Figure 5.1: cwnd dynamics of TCP variants 120
Figure 5.2: RTT dynamics of TCP variants 121

Figure 5.3: E2ED of TCP variants vs BSM intervals 122
Figure 5.4: Average Throughput of TCP variants vs BSM intervals 122
Figure 5.5: Total Dropped Packets of TCP variants vs BSM intervals 125
Figure 5.6: PDF of TCP variants vs BSM intervals 125
Figure 5.7: BSM Sent Packets in the presence of TCP variants 127
Figure 5.8: BSM Received Packets in the presence of TCP variants 127
Figure 5.9: BSM Received Packets in the presence of TCP variants and No. of nodes 127
Figure 5.10: E2ED of BSM in the presence of TCP variants 128
Figure 5.11: Average Throughput of BSM in the presence of TCP variants 128
Figure 5.12: PDF of BSM in the presence of TCP variants 128
Figure 6.1: cwnd dynamics of TCP variants 140
Figure 6.2: RTT dynamics of TCP variants 141
Figure 6.3: E2ED of TCP variants vs BSM intervals 141
Figure 6.4: Average Throughput of TCP variants vs BSM intervals 141
Figure 6.5: Dropped Packets of TCP variants vs BSM intervals 141
Figure 6.6: PDF of TCP variants vs BSM intervals 141
Figure 6.7: BSM Sent Packets in the presence of TCP variants 145
Figure 6.8: BSM Received Packets in the presence of TCP variants 145
Figure 6.9: BSM Received Packets in the presence of TCP variants and No. of vehicles 145
Figure 6.10: E2ED of BSM in the presence of TCP variants 145
Figure 6.11: Average Throughput of BSM in the presence of TCP variants 145
Figure 6.12: PDF of BSM in the presence of TCP variants 145

List of Abbreviations

AC : Access Category
ACK : Acknowledgment
AIFS : Arbitration Inter-Frame Space
AIMD : Additive Increase and Multiple Decrease
AODV : Ad hoc On Demand Distant Vector
AP : Access Point
ASV : Advanced Safety Vehicle Program
BER : Bit Error Rate
BPSK : Binary Phase Shift Keying
BSM : Basic Safety Messages
BS : Base Station
BWE : Bandwidth Estimate
CA : Congestion Avoidance
CAM : Cooperative Awareness Message
CAVENET : Cellular Automaton based Vehicular NETwork
CCH : Control Channel
CCW : Cooperative Collision Warning
CityMob : City Mobility
CSMA/CA : CarrierSenseMultipleAccessCollision/Avoidance
CTP : Cabernet Transport Protocol
CVIS : Cooperative Vehicle-Infrastructure Systems
CW : Contention Window
CWND : Congestion Window
DCF : Distributed Coordination Function
DOT : Department of Transportation
DSDV : Destination Sequenced Distance Vector
DSRC : Dedicated Short Range Communication
DSR : Dynamic Source Routing
EDCA : Enhanced Distributed Channel Access
E2ED : End to End Delay
EEBL : Electronic Emergency Brake Light
ESS : Extended Service Set
ETP : Encounter Transfer Protocol
FARE : Frequency Adjustment with Random Epochs
FCC : Federal Communication Commission
FTP : File Transfer Protocol
GloMoSim : Global Mobile Information. Systems Simulation
GPS : Global Positioning System
HSTCP : TCP-HighSpeed
IBSS : Independent Basic Service Set
IEEE : Institute of Electrical Electronic Engineering
IFQ : Interface Queue
IPV6 : Internet Protocol Version 6

ISI : Inter Symbol Interference
IT : Information Technology
ITS : Intelligent Transportation System
LoS : Line of Sight
MAC : Media Access Control
MANET : Mobile Ad hoc Network
MBWA : Mobile Broadband Wireless Access
MCTP : Mobile Control Transport Protocol
MIB : Management Information Base
MIMD : Multiplicative Increase Multiplicative Decrease
MOCCA : Mobile Communication Architecture
MOVE : Motor Vehicle Emission Simulator
MSS : Maximum Segment Size
NAM : Network Animator
NCTUns : National-Chiao-Tung-University Simulator
NLoS : Non-Line of Sight
NOW : Networks on Wheels
NS : Non-Safety
NS3 : Network Simulator
OBU : On Board Unit
OCB : Outside Context of BSS
OFDM : Orthogonal Frequency-Division Multiplexing
OLSR : Optimized Link State Routing
OMNeT++ : Objective Modular Network Testbed in C++
OPNET : Optimized Network Engineering Tool
OWD : One-Way Delay
PBC : Periodic Broadcast
PDR : Packet Delivery Ratio
PDF : Packet Delivery Fraction
QAM : Quadrature Amplitude Modulation
QoE : Quality of Experience
QoS : Quality of Service
QPSK : Quadrature Shift Phase Keying
RERR : Route Error
RREQ : Route Request
RREP : Route Reply
RSS : Received Signal Strength
RSU : Roadside Unit
RTO : Re-Transmission Timeout
RTT : Round Trip Time
RWND : Receiving Window
SACK : Selective Acknowledgement
SCH : Service Channel
SCHI : Service Channel Interval
SCTP : Stream Control Transmission Protocol
SNR : Signal to Noise Ratio
SOWD : Smoothed One-Way Delay

SS	:	Service Set
STA	:	Station
STCP	:	TCP-Scalable
STRAW	:	STreetRAndom Waypoint
SUMO	:	Simulation of Urban Mobility
SVA	:	Slow/Stopped Vehicle Alert
TCP	:	Transmission Control Protocol
TCPW	:	TCP-Westwood
TMC	:	Traffic Management Centre
UDP	:	User Datagram Protocol
UTC	:	Universal Coordinated Time
VANET	:	Vehicular Ad-hoc Network
VanetMobiSim	:	VANET Mobility Simulator
V2I	:	Vehicle to and/or Infrastructure/Infostation
V2II	:	Vehicle to Infrastructure Integration
V2V	:	Vehicle to Vehicle
V2X	:	V2V and V2I
VoIP	:	Voice Over Internet Protocol
VSC-A	:	Vehicle Safety Communications – Applications
W	:	Window
WAVE	:	Wireless Access for Vehicular Environment
WBSS	:	WAVE Basic Service Set
WiMax	:	Worldwide Interoperability for Microwave Access
WSA	:	Wave Service Advertisement
WSM	:	WAVE Short Message
WSMP	:	Wave Short Messaging Protocol

Chapter 1

Introduction

1.1 An Overview of DSRC

Intelligent Transportation Systems (ITS) are being increasingly and widely deployed in the leading nations of the world, in order to bring significant improvement in transportation system performance including, *inter alia*, reduced congestion, improved safety and increased traveller convenience. Advances in Information Technology (IT) have made it possible to enable entities within ITS system, such as vehicles, roads, traffic lights, and message signs, to become smart by incorporating microchips in them along with sensors, thereby empowering them to communicate with each other through wireless communication technologies.

VANET is an integral component of ITS and is a distinctive class of Mobile *Ad hoc* Network (MANET), which is characterised by unpredictable rapid topology changes due to high mobility of stations (STA). This renders the applicability of the protocol suite of MANETs unsuitable for VANETs. A VANET can be established by using several technologies [1][2] that include Wi-Fi, WiMax, Cellular, Infrared, Global Positioning System (GPS) and Bluetooth, with the selection based on the application requirements. The prime objective of most vehicular wireless technologies is to provide additional layer of safety for both pedestrian and on-board passengers including, of course, the driver.

Each technology comes with its own set of strengths and weaknesses in terms of usage cost, efficiency and effectiveness. Wi-Fi based DSRC (IEEE802.11p) using the 5.8 or 5.9 GHz spectrum band is only a draft standard but is considered well suited for VANET communication, because of its mandatory use with telematics devices aboard the vehicle. The innate characteristic of joining devices to the network without going through conventional authentication and association procedure makes DSRC suitable for vehicular safety messages. Due to these reasons, DSRC standards have become the heart of VANET communication [3].

DSRC has attracted increased interest in both the research community and industry circles, and it is widely recognised as an emerging and promising technology for VANETs. The core motivation to design and develop DSRC standard is to provide guaranteed delivery of time critical safety services to prevent vehicle-related accidents on the road [4]. Additionally, non-safety infotainment type of applications, such as electronic toll collection and roadside service information, can also be provisioned [5]. Currently, there is no technology available wherein both safety and non-safety applications can seamlessly coexist. Research efforts are afoot in each layer of DSRC protocol stack to enhance communication efficiency and performance effectiveness in all possible ways for both classes of applications.

Application	**Layer 7**	**IEEE 1609 WAVE**
Presentation	**Layer 6**	
Session	**Layer 5**	
Transport	**Layer 4**	
Network	**Layer 3**	
Data Link	**Layer 2**	**IEEE 802.11p ASTM2213 DSRC**
Physical	**Layer 1**	

Figure 1.1: The distinction between WAVE & DSRC as per OSI model

DSRC and WAVE are interchangeable terms which use IEEE P1609x along with IEEE 802.11p communication standard. However, there are certain general technological differences between them, as shown in Figure 1.1. The IEEE P1609x covers upper layers 3-7 protocols dealing with content delivery, management and security related issues [6][7]. The IEEE802.11p is the amendment of 802.11a WLANs Media Access Control (MAC) and PHY layers, which specifically supports vehicle mobility at speeds up to 150 km/hr with a maximum RF range of up to 1 km radius [8]. The changes appear in frame format, with increase in delay spread tolerance. The purpose of amendment was to evolve new vehicular wireless communication standards that seamlessly execute safety related critical applications on time.

The communication in DSRC based VANET can take place in three different modes: Vehicle to Vehicle (V2V) in *ad hoc* mode, Vehicle to Infostation (V2I) in infrastructure mode, and thirdly, a combination of the two in a hybrid fashion (V2X). The single pass-through communication link with an infostation or vehicles can exist only for a transient period of time because of high velocity of vehicles. The amendments made in MAC and PHY further define a way to exchange data between V2X using short-lived communication links in an infrastructure-less network model. Keeping short network session in view, the DSRC standard tends to have spontaneous connection between devices to meet time sensitive application requirements. To ensure this, DSRC devices use wildcard Basic Service Set ID (BSSID) having a value of all 1's in the frame header, and as a result, On Board Unit (OBU) begins exchanging fail-safe safety messages the moment it senses RF signal.

DSRC standard mainly targets periodic and event driven safety services. Besides safety, it also has architectural features and technical capability to provide a range of real life non-safety applications catering for mundane information and some kind of entertainment. So, based on V2V, V2I or V2X type of communication, VANET applications can be broadly classified into Safety and Non-safety ones [9][10].

1.2 DSRC Safety Messaging Services

DSRC uses WSMP at transport layer to send safety messages known as Basic Safety Messages (BSM) or DSRC messaging, usually very short about 100 - 300 bytes in size. It usually demands infrastructure-less V2V communication due to highly dynamic network topology and strict low latency constraint. Two kinds of safety services are provided: one is periodic broadcast and the other event driven. Periodic broadcasting is similar to beaconing, and it is one of the fundamental and high priority services in DSRC network. Periodic messages are sent regularly by all vehicles to inform neighbours about their current status, such as vehicle geo-position, speed, direction of movement etc. In contrast, event driven messages are those which are sent only during hazardous or abnormal situations if and when they arise.

In order to establish an efficient and effective DSRC system, the routine safety messages have got to be delivered both in a timely and reliable manner. To meet these requirements, BSM requires periodic broadcast at the average interval of 0.05s – 0.15s.

The acute need for safety information usually arises at points in this time interval when the vehicles are in close proximity to the sender for whom the probability of sudden collision-like event is high. Safety applications such as pre-crash sensing and alerting driver in advance for any potent danger ahead, like car-to-car collision avoidance, dangerous lane changing warning etc. are aimed at minimizing road accidents by using traffic monitoring and management applications.

Several projects, like Vehicle Safety Communications – Applications (VSC-A) [11], PReVENT [12], Cooperative Vehicle-Infrastructure Systems (CVIS) [13], COOPERS [14], Networks on Wheels (NOW) [15] and FleetNet [16] have come up with many draft recommendations for a variety of safety applications for DSRC. A few among them include Cooperative Collision Warning (CCW), Electronic Emergency Brake Light (EEBL), and Slow/Stopped Vehicle Alert (SVA).

1.3 DSRC Non-safety Application in VANET

Non-safety commercial applications enable passengers to access infotainment and traffic efficiency services like Internet access, interactive communication, online games, electronic payment services, downloading content and information updates etc. to make their trips more comfortable. The notable difference between safety and non-safety application is that the safety applications are capable of sending and processing messages in real-time [17][18]. The driver and passengers can seamlessly access both kinds of services from Base Station (BS) or from another vehicle using wireless access technologies [19]. DSRC works on wireless LAN model and there are various previous researches made on IEEE 802.11 based infotainment services over VANET. Jorg Ott and Dirk Kutscher [20][21] experimented with V2X communication using TCP over IEEE 802.11b. They found that 802.11b supports persistent Internet connection on slow moving vehicles with high data rate. It has the capability to intermittently connect fast moving vehicles at the approximate speed of 40 - 180 km/h with data rate suitable for applications which do not require continuous connection like sending an email message, browsing a web page etc.

Various types of service applications have also been developed to ease traffic bottleneck such as electronic toll collection, congestion charging, information provision etc. [2][22]. Advertising messages on the fly is also an important service application, which when combined with GPS can alert the driver or passenger to the availability of local services and facilities. People want to stay connected to the world using Internet while on the move using devices like Tablet, Laptop, Mobile Phone and Telematics. Car manufacturers are starting to offer such capabilities by seamless integration with these gadgets, and the hardware infrastructure is already in place. Examples of comfort application are on-board Internet, platooning wheel gaming, roadside service information, V2V webcast etc [23][24]. All these infotainment-based applications can be used on V2X (V2V/V2I) communication. TCP is the common choice for designing non-safety application but using TCP on VANET raises certain serious issues that impinge negatively on the existing safety message communication. This thesis attempts to resolve them in a comprehensive way.

1.4 Research Motivation

Similar to office and home networks, and perhaps more so, high bandwidth is required for communication with moving vehicles for various purposes, such as fast web browsing, online gaming, fast downloading of large files containing huge data, video and pictures/images. Also, applications like video conferencing, VoIP, webcast and remote database connection consume huge bandwidth. These kinds of delay-insensitive or delay-tolerant applications can certainly provide value added services to the end user using conventional transport layer protocol TCP.

As the architecture of TCP is highly tuned for wired networks, it performs, of necessity, poorly in vehicular wireless network environment due to congestion, channel fading, multi-path propagation and frequent disconnections. To overcome these hurdles, substitutes of TCP such as Stream Control Transmission Protocol (SCTP) [25], Encounter Transfer Protocol (ETP) [26], or Cabernet Transport Protocol (CTP) [27] have been developed whose performance is better in dynamic topological network. But, unfortunately, they cannot be deployed here because of two most important reasons *viz.* proprietary issues and backward compatibility problems.

To circumvent these difficulties and to address associated issues, this thesis attempts to find a workable and viable scheme that proceeds by:

- Determining an optimal safety message interval rate over a reliable modulation scheme,
- Identifying a most suitable conventional TCP congestion control mechanism that can behave in a friendly way with BSM application,
- Studying the effect on BSM when used with TCP-based applications, and finally
- Modifying conventional TCP suitably to conform with application and communication protocol requirements while maintaining optimal throughput for BSM and without hurting its priority status.

If seamless co-deployment of BSM and delay-insensitive types of applications could be achieved cost-effectively, under the rubric of DSRC technology umbrella, the society at large stands to benefit immensely and significantly by way of enhanced quality of life. Hopefully, those days are not a long way off.

1.5 Challenges of Deploying TCP in VANET

One needs to have TCP/UDP for delay-insensitive data delivery over IEEE 802.11p communication technology. TCP is used for reliable delivery of data traffic over guided media; however, it has acceptable performance only in static wireless network. The performance takes a huge hit, if TCP is used for file transfer application over frequently disconnected network. DSRC uses IEEE P1609.4, which is based on Enhanced Distributed Channel Access (EDCA) of IEEE 802.11e for multi-channel operation, where dedicated control channel has high priority to deal only with BSM. The purpose or utility of having multi-channel is called into serious question or may very well look meaningless if the device has got only one single radio; moreover, multi-radio devices bring in a whole lot of other kinds of complexities. Experiments indicate that if flow of safety and non-safety traffic were constrained to flow over a single radio device having a single wireless channel, then ensuring fair flow of this *mixed* traffic gets very difficult, and challenging too, and therefore this becomes an important and even a critical issue needing urgent solution.

For TCP based infotainment applications, high index 64-Quadrature Amplitude Modulation (64-QAM) scheme having a maximum payload of 27 Mbps, is required for good performance; but at the same time one has to consider its impact over BSM and messaging interval. Through experiments [28], it has been well proved that higher modulation underperforms when V2V or V2I communication distance is long.

When one plans to use TCP over DSRC it is impossible to ignore or disregard integral periodic BSM broadcast which is, after all, the *raison d'être* of VANET. The efficient use of BSM application is critical for preventing road accidents. The broadcasting rate of time sensitive, critical BSM depends upon the importance of the application. Basic safety messages like the one for collision avoidance absolutely requires strict low latency broadcast interval rate to prevent potential dangers and possible accidents.

The unsuitability or misfit of TCP arises mainly because BSM applications are small in size but are broadcast periodically, whereas TCP based applications are unicast on demand and have bigger packet size. Periodic BSM broadcasting invariably occupies a large portion of the channel bandwidth due to flooding of safety messages or periodic updating of routing tables [29][30].

The load capacity of a wireless network is limited to the available channel bandwidth and the number of STAs that share the channel [31][32]. When the network density increases, the capacity available to each STA naturally is reduced. The channel congestion arises when the channel utilization at each STA exceeds the capacity, which results in performance degradation with high packet loss and latency.

The safety *vs* non-safety channel bandwidth usage scenario is displayed in Figure 1.2. The availability of bandwidth for infotainment services depends upon the frequency of BSM traffic. Theoretically, higher vehicular density increases BSM broadcast traffic on the channel, and as a result, the available time slot for non-safety decreases [33]. Figure 1.2 (a) shows a sparse vehicular density scenario where BSM broadcast is very low; as a result, some of the bandwidth remains unused. Of course, under the current protocol, the unutilized bandwidth of DSRC can be put to good use for infotainment applications. But, how to use that unutilized bandwidth in a fair and balanced manner is a major challenge in TCP design. This thesis looks at this issue carefully and an attempt is made to provide a satisfactory solution.

Figure 1.2 (b) shows the average vehicular density and the available channel slot for non-safety services; however, in Figure 1.2 (c) the non-safety time slot has been squeezed to a great extent, but can still be used with low throughput applications.

Another possibility is depicted in Figure 1.2 (d), where the time slot for infotainment is too short, making its use difficult for commercial non-safety applications, but still feasible for use-cases like electronic toll collection or local services information. According to DSRC philosophy, BSM constitutes the core application and other non-safety communication necessarily takes a back seat. This is clear from Figure 1.2 (e), where the time slot available for non-safety application use is nearly negligible.

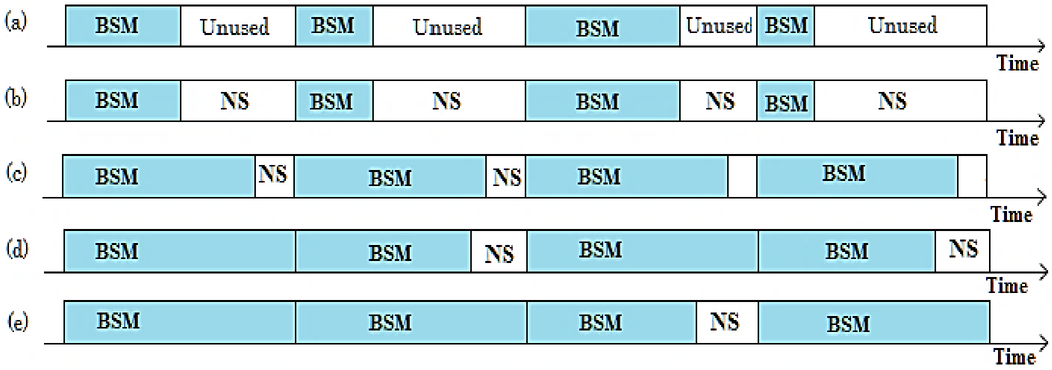


Figure 1.2: Safety and Non-Safety (NS) channel usage scenario

Figure 1.2 shows Safety and Non-Safety (NS) channel usage scenario; (a) typical unused bandwidth, (b) average available bandwidth for NS, (c) reduced bandwidth for NS, (d) least bandwidth available for NS and (e) most of the bandwidth occupied by BSM.

Most of the commercial services are accessed through roadside gateway, so it requires Vehicle to Infrastructure Integration (V2II) in a single or multi-hop manner. Since the capability of vehicles to perform essential BSM is considerably weakened in such circumstances, how to provide BSM service effectively and efficiently along with commercial infotainment services and utility-oriented services in DSRC has become an active area of research. Further, other more fundamental problems come to the fore: how much of DSRC’s bandwidth can really be made available for non-safety applications use, and whether it will be feasible at all to do IP like communications without affecting the reliability of existing BSM communications.

So far there does not seem to exist any satisfactory current study that considers seriously all the possible ill-effects on safety applications, while using TCP based non-safety application over DSRC. Devising a way to deploy TCP over 802.11p, without significantly affecting the reliability of existing BSM applications, is the main challenge that is taken up in this work.

Following are some of the challenges and issues facing TCP when deployed over DSRC:

- Infotainment related services require high data throughput for good performance. DSRC over 10 MHz of channel can have a maximum of 27 Mbps data rate, which is sufficient to serve bandwidth-hungry delay-insensitive applications. But, when inter-vehicular distance gets large, higher modulation schemes inflict severe damage on the performance of safety applications. One challenge, therefore, is to find the optimal modulation scheme at the appropriate inter-vehicular distance that provides acceptable performance.

- Over the reliable modulation scheme and optimum message interval, under V2X transmission, the TCP negatively impacts on periodic broadcast of BSM, in terms of the most appropriate practical metrics such as throughput, E2ED, and PDF.

- One of the TCP problems in vehicular wireless network is that it interprets transmission errors as congestion situation and reduces the throughput by frequently invoking slow-start congestion control mechanism. There are various congestion control algorithms available to ameliorate this problem; yet TCP continues to provide poor performance in the VANET scenario.

1.6 Research Contribution

To meet the above-said challenges and issues upfront, the present research study was undertaken, and it is wholly simulation-driven. The following findings and outcomes of the investigation constitute the contribution to this important and growing VANET area of research.

- The most appropriate modulation scheme for BSM over varying vehicular density is determined by simulation.

- The most appropriate broadcast interval rate is fixed for BSM over reliable modulation scheme at varying vehicular density, through simulation studies.

- Performance analysis of file transfer application using various TCP congestion control algorithms is made. Also, the impact of these congestion control algorithms on BSM is systematically studied.

- The performance of the TCP-Scalable algorithm is critically examined and an optimized version named vScalable-TCP, best suited to VANETs, is designed. This enhanced congestion control algorithm performs better for infotainment applications, without harming existing BSM flows.

- The performance of the TCP-LP algorithm is carefully studied, with a view to testing how far the success of slow-start congestion control algorithm, found so effective in vScalable-TCP, can be replicated here. This has led to the formulation of another algorithm named vLP-TCP, which offers superior performance without much hampering the existing BSM flows.

The research methodology adopted in this study is based exclusively on experiments by computer simulation. The modelling process entails a comprehensive appraisal of DSRC safety messaging application as well as TCP congestion control algorithms with their characteristic features. The design process demands a thorough understanding of DSRC concepts, a careful examination of safety short message applications and a fresh in-depth re-look at the internals of TCP congestion control algorithms. The general principle has been adopted for transmission parameters in order to optimise congestion control algorithms of TCP that do not interfere much with safety application. The implementation section describes construction of the optimized congestion control algorithms and conducting of experiments on them, in a computer based simulated environment. The research challenges are addressed by proposing an efficient and effective approach for TCP that allows mitigation of fairness problem in DSRC, and by critically analysing the simulation results. The novelty in this approach lies in cleverly tweaking TCP slow-start congestion control algorithm *vis-à-vis* BSM message exchange rate.

1.7 Dissertation Organization

The remainder of the thesis is organized as follows:

Chapter 2 is divided into two sections. In the first section, the author provides an overview of MANET, VANET, followed by a discussion on DSRC including its architecture, and then TCP in VANET and TCP congestion control mechanism. In the next section, previous contributions available in the literature, pertaining to performance analysis on different modulation schemes, BSM broadcast rate, and accessing TCP based application in VANET are reviewed.

Chapter 3 sets the stage for conducting simulation experiments and defines the performance analysis metrics. Then, it discusses research work done with NS2 simulation tool, and the need for traffic mobility simulator MobiSim and Nakagami propagation model in the context of VANET. Finally, it defines and describes two simulation scenarios to be used in the subsequent chapters.

Chapter 4 comprises three sections. In section 4.1 performance analysis of safety short messaging application is carried out, over four different modulation schemes and varying message intervals. In section 4.2, the performance of many TCP congestion control variants are studied. Section 4.3 discusses the impact of TCP congestion control algorithm over safety short messaging application, using reliable broadcast interval and optimal modulation scheme.

Chapter 5 discusses optimization of TCP congestion control algorithm that can work non-invasively with safety applications, providing along the way some efficiency for non-safety applications. Based on the outcome of previous TCP performance analysis, TCP-Scalable congestion control algorithm has been chosen for optimization. A newly formulated improved TCP-Scalable algorithm, named here as vScalable-TCP, is shown to provide a fair share of bandwidth to safety application and is found to be suitable for deployment over VANET.

Chapter 6 discusses another optimized TCP congestion control algorithm that also works in a friendly way with safety applications. It has been decided to work on TCP-LP because certain characteristics it bears make it suitable for deployment over VANET.

We have formulated an optimized TCP-LP algorithm keeping in mind the high priority nature of safety application and named it as vLP-TCP.

Chapter 7 summarizes the main findings of this research, lists potential research issues for future work, and points out some limitations of the present study.

Chapter 2

Background Study and Related Work

This chapter is divided into two parts. In the first part, an introduction to MANETs and VANETs, and their detailed characteristics and differences are given, highlighting the wireless communication technology suitable for VANETs. The DSRC wireless standard used in VANET is discussed, including its key architectural components such as PHY layer, MAC layer and P1609.x. In the second part, a survey of related works is given, focusing on different modulation schemes used in DSRC, message broadcast interval rates, and finally implementation of TCP over VANETs.

2.1 MANET

A Mobile *Ad hoc* Network (MANET) is a group of interconnected wireless devices that are formed on the fly without the use of any infrastructure. MANET is an autonomous system of self-organizing, self-configuring and self-controlling network. In MANETs, the nodes move in an arbitrary and random fashion, and these characteristics make the network topology totally unpredictable. Signal range of MANET devices covers only a few meters because of hardware limitations. Since a MANET node has limited signal range, it is also made to act as a relay to forward data to farther, out-of-range nodes via intermediate nodes. MANETs can be formed using wireless devices such as mobile phone, laptop and PDA. MANETs are deployed in various contexts, including disaster recovery, emergency situations, on the fly communication establishment for military battle ground and industry among many others [34].

The characteristics of MANETs are quite different from both wired and other types of wireless network as can be seen in Figure 2.1. The salient features of MANETs [34][35][36] are set forth below:

- **Infrastructure-less Communication:** MANETs are formed without the use of pre-fixed or predetermined BS. The network between two or more nodes is promptly created in *ad hoc* mode, which is suitable for various purposes including emergency like situation.

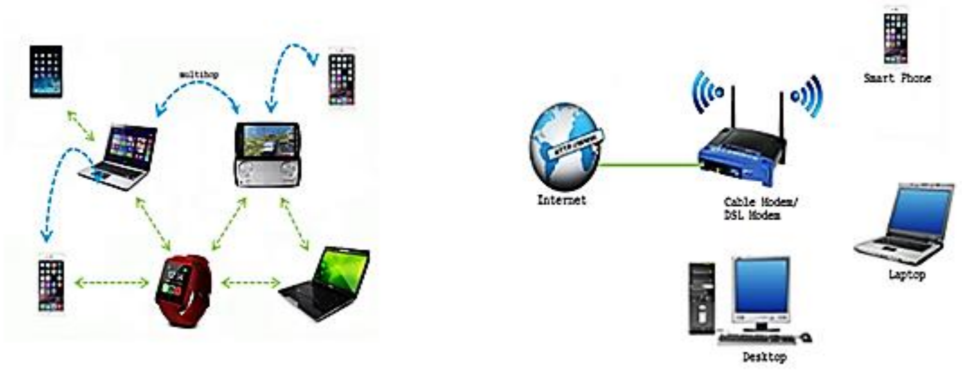
- **Dynamic Topology:** Nodes in MANETs are free to move arbitrarily and randomly; as a result, connectivity or link between the nodes changes over movements. This dynamic flipping of links, changes the network topology fluidly and unpredictably. Dynamic topology is the most distinguishing characteristic of MANETs as compared to other wireless networks.

- **Multi-hop Communication:** In out-of-range wireless transmission conditions, the communication between two distantly located nodes in MANETs takes place in a multi-hop fashion, using several intermediate nodes as routers or relays to forward packets.

- **Self-Configuring:** A node in a MANET can move from one location to another occupied by continuously changing peers. Node movement is possible with the use of self-configuring feature that makes nodes responsible for dynamically discovering other nodes to communicate with or to handle network re-configuration.

- **Distributed Control:** Since MANETs are formed on-the-fly and nodes are autonomous in nature, there is no predefined central management control of the network. The management control work instead is performed by individual nodes in a distributed manner. All the network activities including topology discovery, routing and message delivery, are performed by nodes themselves.

- **Limited Resources:** In most cases, mobile devices in MANETs, such as PDAs, mobile phones, laptops and other wearable devices, have limited capacity for storage, memory and power.



a) A Typical MANET architecture

b) An infrastructure wireless network

Figure 2.1: An example of MANET and Infrastructure wireless network

2.2 VANET

Recent advances in vehicular and road traffic industry have enabled use of sensors, telematics devices and other gadgets to make journeys safer, comfortable and enjoyable as well. To enjoy this facility, new technologies for wireless communications such as VANET have been enlisted, which can be considered as a form of advanced MANET, as it is instantaneous, mobile and uses moving vehicles as nodes in the network [37]. Moving vehicles in a VANET act as either a node or a relay to exchange messages between vehicles, or a road side BS. Typically, a VANET can connect vehicles within a communication range of 100 to 1000 meters when using the IEEE 802.11p communication standard. The acronym VANET is self-explanatory i.e. it is an infrastructure-less communication network for vehicles; however, in reality, it is a hybrid architecture consisting of both V2V and V2I communication, commonly referred to as V2X communication network. The straight V2V is an *ad hoc/* infrastructure-less communication mode whereas V2I relies on pre-fixed infrastructure like roadside BS.

It is essential to remember always that the supreme goal of VANET is to improve road safety and provide a comfortable travel experience for drivers and passengers alike [4][38]. There have been numerous research initiatives, such as COOPERS, CVIS, SAFESPOT, PReVENT, WAVE and Advanced Safety Vehicle Program (ASV) [17], carried out across Europe, US and Japan, to make ITS a reality. VANET is an integral component of ITS which supports both real-time safety applications, and non-safety delay-tolerant infotainment applications. Safety applications, such as collision avoidance, pre-crash sensing or lane changing, are aimed at minimizing road accidents by using traffic monitoring and management applications. Non-safety applications enable passengers to access various services, like Internet access, interactive communication, online games, payment services and information updates, while vehicles are on the move. The key distinguishing feature of safety applications is their real-time operation with strict low latency time constraints [18].

VANET and MANET share many similar features, such as self-organising, self-configuring, self-healing, dynamic topology, multi-hop data transmission, distributed architecture, short range connectivity and omni-directional broadcast. Table 2.1 shows key MANET and VANET characteristics. In both the networks, mobile nodes are able to relay data to the destination in a single hop or multiple hops by themselves. However, there are some notable differences between them. Since the vehicles are moving along the road in a particular direction, the mobility of nodes in VANET are predictable unlike MANET [39]. Furthermore, there is no limitation of storage, processing capability or battery power of nodes in a VANET. Due to fast movement of the nodes, the wireless network topology becomes highly dynamic and unstable. In addition, network density varies significantly over time and location [40].

Characteristic	**MANET**	**VANET**
Signal Range	Short	Short to Medium
Communication Types	Single/Multihop	Single/Multihop
Network Type	Infrastructure-less	Infrastructure/less
Mobility	Unpredictable	Predictable
Topology	Dynamic	Highly Dynamic
Network Management Control	Distributed	Distributed
Resources Including Power	Limited	Unlimited

Table 2.1: Key characteristics of MANET and VANET

Typically, a VANET consists of four major components: vehicles, sensors such as GPS enabled telematics devices, road-side infostation, and Traffic Management Centre (TMC) [41][42]. All these components communicate using wireless communication protocols that determine various aspects of communication such as transmission range, data rate, latency and security. Indeed, timely delivery of data is considered as the key challenge due to rapid topology change, frequent signal disruption, and small contact opportunities in VANET [43].

There are various vehicular wireless technologies available that help regulate traffic flow, provide safety and other useful services. A VANET could use multiple wireless networking technologies including WAVE, DSRC-802.11p, IEEE P160x, WiMax-IEEE 802.16, Bluetooth-IEEE 802.15.1, MBWA-IEEE 802.20, Infrared and cellular, to facilitate communication among the vehicles [19][44].

Table 2.2 shows a comparative study of wireless technologies suitable for VANET. A particular VANET communication technology is elected based on the types of applications that are required to be deployed, mobility support, cost, data transfer rates and signal range. Cellular based LTE is the most suitable technology for high speed broadband communication; however, for fail-safe safety applications, DSRC standard is indeed the best choice.

VANETs are spontaneously formed and have some very interesting features. Following are some of them [8][45]–[48]:

- **Mobility:** In contrast to a MANET, the vehicular mobility pattern in a VANET varies with respect to traffic environment. Rural routes have the lowest vehicular mobility, whereas on a freeway the velocity may reach 120 mph. Also, the vehicular movement patterns are restricted by road lanes and traffic signals, with permissible movement in opposite directions if it is a two-way road.

- **Highly Dynamic Topology:** Even as topology rapidly changes, this mobility is somewhat predictable, unlike a MANET having random nodes movement. Nodes in VANET join and leave the network very frequently, and this feature also changes the topology.

- **Frequently Disconnected Network:** Dynamic topology causes frequent disconnection of communication among the vehicles and Roadside Units (RSUs), especially when they are exchanging information at freeways, where the vehicular densities are relatively sparse. Such frequent disconnection forms a ‘nearly-net’ kind of communication, where certainty of data delivery is always not assured.

- **Lossy Wireless Communication:** At higher vehicular speeds, the wireless channel becomes unsteady and lossy, resulting in signal attenuation, signal interference, packets being discarded frequently, packet collision, data delivery delay, channel fading and Doppler Effect.

- **Storage and Power:** Unlike MANET, VANETs do not suffer from storage and power limitations, since modern telematics devices are embedded with high storage capacity for various purposes and continuously receive power from the vehicle itself.

- **High Processing Speed:** Since storage and power is not a constraint for VANET, the OBUs can enjoy high processing speed using latest computational technology to fulfil the needs for various applications, both time-sensitive safety and delay-insensitive non-safety infotainment ones.

- **Scalability:** A VANET, being essentially *ad hoc* in nature, can grow extensively throughout the road network. This makes VANET protocols susceptible to many wireless channel collisions and signal interference between mobile stations.

Figure 2.2 shows a typical VANET topology in three different configurations: V2V, V2I and V2X. In order to create an information exchange network, each RSU creates a cell consisting of a number of moving vehicles, wherein each vehicle functions as a router to relay signals to farther ones, thus making for a scalable and robust communication system.

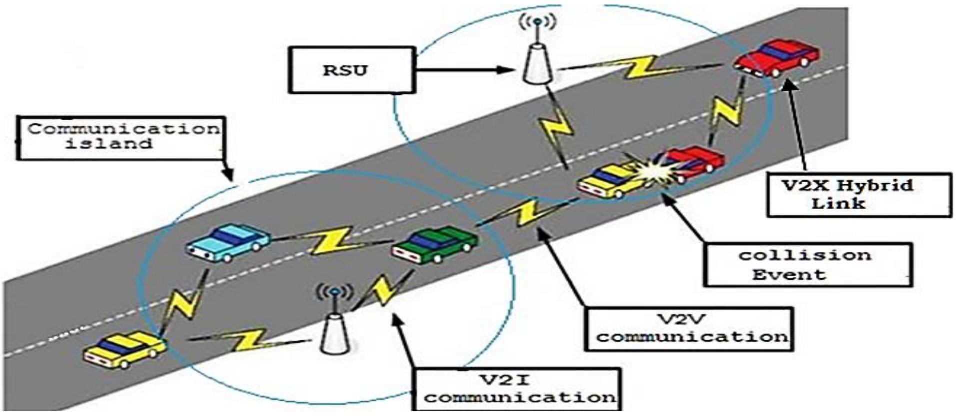


Figure 2.2: V2V, V2I and V2X Communication [49]

2.3 The DSRC for VANET

DSRC is a short-to- medium range communication technology that supports both V2V and V2I communication. In October 1999, the U.S. Federal Communication Commission (FCC) allocated and reserved a total of 75 MHz of DSRC spectrum in the 5.85-5.925 GHz band to be used exclusively for VANET communication. The DSRC protocol suite consists of a set of standards including IEEE802.11p and IEEE P1609.X.

The Intelligent Transportation Society of America (ITSA), a non-profit organisation, recommended using a single standard for the MAC and PHY layers for vehicular network architecture. The FCC adopted this recommendation which is based on WLAN - IEEE802.11 architecture [50]. The IEEE task group modified 802.11 to support and suit the needs of vehicular wireless network environments. The outcome of this work was standardised as IEEE802.11p [51]. Another team of IEEE, known as WG1609, undertook the task of developing the upper layer specifications to append to the DSRC protocol suite. Currently, IEEE 1609 standard consists of four trial specifications IEEE 1609.1 [52], IEEE 1609.2 [53], IEEE 1609.3 [54], and IEEE 1609.4 [55]. Both IEEE802.11p and IEEE1609.x constitute the core of DSRC standard and are collaboratively used to facilitate the provision of WAVE. And this is why DSRC is also known as WAVE.

The primary motive for adopting DSRC is to deploy safety related applications to prevent collision and ease traffic flows; but these days a broad range of non-safety commercial types of applications are also of interest to various groups of people. To date, of the various vehicular wireless technologies available, DSRC is attracting many researchers because it is considered, by far, the most reliable and suitable upcoming technology specifically for V2V collision prevention type of applications. Looking at the technology comparison in Table 2.2, it is evident that DSRC is most suitable in terms of cost, data transfer rate, signal range and suitability for safety applications. The U.S. Department of Transportation (DOT) has roughly estimated that DSRC based communication can help eliminate or at least reduce up to 82% of all crashes or accidents on the US roads, thereby potentially saving thousands of lives, and of course money as well [8].

Standard	**Frequency Band in GHz**	**Data Transfer Rate in Mb/s**	**Approximate Outdoor Signal Range in Mtr.**	**Mobility support in ≈ km/h**	**Suitable for Outdoor Network**	**Suitable for Safety Application**	**Suitable for Non-Safety Application**	**Access Cost**
IEEE 802.11a	5.1-5.8	25-54	45	40-120	Low	No	No	£
IEEE 802.11b	2.4-2.5	11	100	40-150	Low	No	No	£
IEEE 802.11g	2.4	6-54	140	40-120	Low	No	No	£
IEEE 802.11n	2.4/5	254	250	40-120	Low	No	No	£
IEEE 802.11p (DSRC)	**5.8-5.9**	**3-27**	**1000**	**40-150**	**High**	**Very Good**	**Possible**	**≈ 0**
IEEE 802.20 (MBWA)	3.5	80	15km	100-250	High	No	Yes	Abandon
IEEE 802.16 (WiMAX)	2.3, 2.5 & 3.5	1-3Gb/s	≈50km	60-250	High	No	Yes	£££
IEEE 802.15.1 (Bluetooth)	ISM band 2.4–2.48	1-24	100	20-30	Very low	No	No	≈ 0
Infrared	835–1035 nm	1-4	150	250	Medium	No	No	£
Cellular	Generation Dependent	≈1 Gb/s	≈50 km	100-250	High	No	Yes	££

Table 2.2: Comparative study of wireless technology suitable for VANET [17]

2.4 Architecture of DSRC

The DSRC system concept is based on IEEE 802.11 Wi-Fi system which provides short range, low latency and high throughput two-way communication. The communication architecture follows OSI reference model as seen in Figure 2.3.

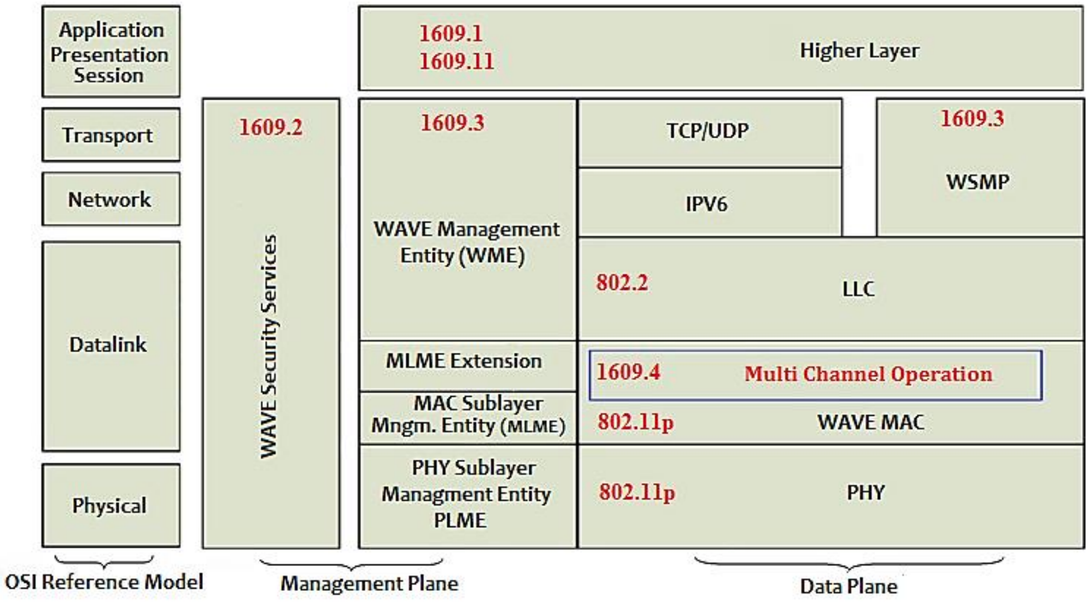


Figure 2.3: Architecture of DSRC

The architecture of DSRC has been so designed as to reduce protocol overhead, making it suitable for real-time applications. Furthermore, the system is designed to support different types of physical media and various types of safety and non-safety applications. The PHY layer architecture of DSRC is in accordance with IEEE 802.11a with some modification as seen in Table 2.3, whereas MAC layer follows EDCA of IEEE802.11e. The other important specification, such as IEEE P1609.x, is designed to facilitate upper layer functions. Envisioning future heavy use of non-safety infotainment kind of applications, the conventional TCP/UDP has also been accommodated at the transport layer in conjunction with WSMP which is dedicated for safety application.

2.4.1 PHY Layer

DSRC is defined in detail in the IEEE 802.11p PHY implementation as recommended by the FCC [56]. This standard uses an Orthogonal Frequency-Division Multiplexing

(OFDM) transmission technique adopted from IEEE802.11a. In Table 2.3**,** the main PHY parameters of IEEE 802.11p and IEEE 802.11a are compared. The DSRC-PHY layer uses a bandwidth of 75 MHz, positioned in the spectrum range of 5.850 - 5.925 GHz. These 75 MHz channels are divided into seven 10 MHz channels each. The central rendezvous Control Channel (CCH) is primarily used for broadcasting, transmission and establishing communication through service channels (SCHs). The rest of the six SCHs are used for both safety and non-safety two-way communication between RSUs and OBUs and between OBUs. Figure 2.4 illustrates DSRC operating channels.

Operating Channel No.	172	174	176	**178**	180	182	184
		175			181		
Channel f in GHz	5.860	5.870	5.880	**5.890**	5.900	5.910	5.920
Channel Purpose	SCH safety	SCH	SCH	**CCH**	SCH	SCH	SCH reserved

Figure 2.4: DSRC/WAVE Operating Channels

The decision to reduce single channel bandwidth to 10 MHz from that of IEEE802.11a, is influenced by the special characteristics of propagation environment. This decision is made essentially to bring down Inter Symbol Interference (ISI) due to multi-path signal propagation.

In order to use multi-channel operation, DSRC provides IEEE P1609.4 protocol at MAC sub-layer. In a single radio device, all the 7 channels cannot be used simultaneously; thus, each mobile station has to constantly switch at an interval of *50ms* between the CCH and the Service Channel (SCH), which is referred as Control Channel Interval (CCHI) and Service Channel Interval (SCHI). To ensure that the requirement of low latency is met, especially when safety data are sent, the switching time should not exceed 100*m*s [57]. OFDM technique, which is used for transmission, divides each channel into several sub-carriers spaced by 0.15625 MHz [58]. OFDM based on 802.11a is used so that existing IEEE 802.11a Wi-Fi chip architecture can be used as the basis for inexpensive DSRC deployment. The amendments made to DSRC-PHY layer *vis-à-vis* IEEE 802.11a are compared in Table 2.3.

Parameter	IEEE 802.11p	IEEE 802.11a
Frequency Band	5.850 - 5.925 GHz	5.15-5.35GHz; 5.725-.835GHz
Allocated Spectrum	75 MHz (US) & 30 MHz (EU)	40 - 20 MHz
Channel Bandwidth	10 MHz/ 20 MHz	20 MHz
Data Rate	3 - 27 Mb/s	6 - 54 Mb/s
Modulation	BPSK, QPSK, 16-QAM & 64-QAM	BPSK, QPSK, 16-QAM & 64-QAM
Code Rate	1/2, 2/3 and 3/4	1/2, 2/3 and 3/4
Subcarriers Number	52	52
OFDM Signal Duration	8 μs	4 μs
Guard Interval	1.6 μs	0.8 μs
FFT Period	6.4 μs	3.2 μs
Preamble Duration	32 μs	16 μs
Subcarriers Spacing	0.15625 MHz	0.3125 MHz
Outdoor Signal Range	< 1000 m	< 100 m
Suitable for Mobility	High	Low
Standards	IEEE, ASTM, ETSI, IETF	IEEE

Table 2.3: Comparison of PHY layer parameters of IEEE802.11p and IEEE802.11a

The presently available wireless technology has the capability to listen to only one channel at a time. In the initial stage of IEEE 802.11p based VANET deployment, there will only be one physical transceiver. The demerit of having a single transceiver is that only one channel can be monitored despite having multi-channel facility. This drawback can be overcome by equipping more than one transceiver device, so that more than one channel can be simultaneously accessed, thereby making effective utilization of bandwidth. In other words, if a vehicle, for example, is equipped with dual radio telematics devices, then simultaneously one transceiver can handle safety related application on CCH, while non-safety applications on SCH can go through the other. In this way reliability is assured and system performance enhanced. However, many challenges crop up in such a scenario, like equipment cost and processing complexity [7]. For experimental purposes, not a single freeware network simulator is available that has full multi-channel capability with single or multiple transceivers.

2.4.2 MAC layer

The MAC of 802.11p must provide reliable and efficient channel access mechanism to support different category of applications. Rapid topology changes and fast movement of vehicles in VANET make it difficult to efficiently share wireless medium. In order to provide timely delivery of delay-sensitive safety messages, the 802.11p MAC simplifies the connection establishment procedure by omitting channel scanning and authentication procedures.

The EDCA of IEEE 802.11e adds Quality of Service (QoS) to 802.11 based networks, which is an upgraded version of Distributed Coordination Function (DCF). The DCF follows Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) channel access method for distributed and decentralized wireless communication. The IEEE 802.11p based mobile nodes will follow contention based EDCA paradigm as a MAC method. Based on EDCA, MAC adopted four different data Access Category (AC) levels with varying transmission priorities. These ACs have different Arbitration Inter-Frame Space (AIFS) and different back-off parameters. A node by following EDCA has to first listen to the channel, and if the channel is free for an AIFS, it will start transmitting frame. And, if the channel is busy or becomes busy during the AIFS period, then the node must perform a random back-off procedure [59][60]. In other words, the MAC-EDCA protocol is a stop-and-wait protocol, and therefore the sender always waits for an Acknowledgment (ACK) to be sure of successful reception of transmitted packet. If a source node does not receive an ACK within a bounded time, a back-off algorithm is invoked before a re-transmission is allowed.

To transmit a packet, the contention window (*CW*) size will be doubled from its initial value (*CWstart*), until a maximum value (*CWend*) is reached. Whenever packets need to be dropped due to reaching maximum number of channel access attempts or upon successful transmission, the *CW* is again set to its initial value [61]. This is due to the fact that during high channel utilization periods, it is convenient to spread the channel access attempts over time.

In a broadcast situation, none of the receiving nodes will send ACKs in response to a source node because there is no RTS/CTS handshaking method applied. This is the reason a sender never knows if all the recipient nodes have successfully received the transmitted packet. It performs at most one back-off procedure similar to initial channel access attempt when a busy channel was sensed. Thus, intended broadcasting packet has to undergo at most one back-off procedure.

2.4.3 IEEE P1609.x

The IEEE P1609.x is a family of standards for WAVE forming upper layer protocol stack. It provides a set of services and interfaces that collectively enable secure vehicular wireless communication. Together, these standards are designed to provide the foundation for a broad range of applications in the vehicular transportation environment, including vehicle safety, automated tolling, enhanced navigation, traffic management and many others. The IEEE 1609 family consists of draft standards 1609.1, 1609.2, 1609.3 and 1609.4.

The IEEE P1609.1 is a WAVE resource manager that is considered to constitute the core part of WAVE architecture. It defines data flows, and command & data storage formats. It also defines types of OBU devices that can be supported, and interaction among OBUs and outside of the OBUs [59]. The IEEE P1609.2 deals with security services for application and management messages. It specifies secure message formats and processing as well as the circumstances for utilizing secure message exchanges. The IEEE P1609.3 defines WSMP which is used for networking services. It defines Network and Transport Layer services that deal with addressing and routing. It is an alternative to IP and deals specifically with periodic delay-sensitive safety message exchange. It also deals with Management Information Base (MIB) for WAVE protocol stack. The goal of P1609.3 is to provide vehicular communication system with two protocols namely TCP/IPv6 and WSMP. WSMP is given higher priority and handles time sensitive communication between vehicles in V2V [62]. The IEEEP1609.4 standard makes the necessary changes to 802.11 MAC to support multi-channel operation in WAVE [63][64] .

2.4.4 Networking Services

The Transport and Network layer of DSRC contains a specific common protocol known as IEEE P1609.3 – WSMP. It is designed to be used only for periodic safety messages that has low overhead and does not require routing when source and destination are one hop away. The WSMP carries a total overhead of 11 bytes, whereas TCP/IPv6 packet 52 bytes. WAVE Service Advertisement (WSA) protocol is also included in this part of DSRC protocol stack [65]. It is a management frame, transmitted on the central CCH and contains information about services found at different SCHs and about safety messaging services. Services may vary from commercial advertisements to any specific requested service of interest. It is envisioned that roadside BS will most likely utilize the WSA distribution; however, OBUs in vehicles are not restricted to offer services such as music streaming to cars in platooning.

WSA allows the applications to directly control the MAC and PHY layer parameters, such as TX power, data rate, channel number and receiver MAC addresses. In order to achieve low latency, WSMP over the CCH skips the steps for forming a WAVE Basic Service Set (WBSS) to deliver IP and WAVE Short Message (WSM) traffic on the SCHs.

The DSRC designers have envisioned potential use of infotainment applications while vehicles are on the move. Keeping this in view, DSRC protocol suite also includes conventional transport layer protocol known as TCP/UDP and network layer IPv6 for accessing several unicast based reliable and ordered delivery of delay-insensitive applications. It facilitates several acceptable delay-tolerant services, with the support of flow control, congestion control and data packet’s integrity. Unfortunately, TCP still performs poorly in highly dynamic and frequently disconnected vehicular networks.

If MAC layer multi-channel media access protocol IEEE P1609.4 is used, then TCP/IPv6 or UDP based application works only on SCH whereas WSMP for short messaging application works on both CCH and SCH to carry high priority safety messages.

2.4.5 Authentication and Association Process in DSRC

In order to establish connection with vehicles or an Access Point (AP) much faster than IEEE 802.11a, there is no authentication and conventional association pairing scheme built into IEEE 802.11p network. The purpose of taking this approach is to minimize device’s joining time to a great extent, which is a must for time critical safety communication. In this way, the DSRC equipped car can send relatively more data to one hop vehicles and RSU than when using IEEE 802.11a protocol. According to Mei-Wen Li et al [66], at the speed of 60 km/hr, the connection duration will last approximately 14 seconds and it is enough to support some TCP-based applications, such as clients’ emails or instant messaging. IEEE 802.11p has less likelihood of consecutive packet loss than with IEEE 802.11a connection. Table 2.4 shows the comparison for VANET connection contact duration for IEEE 802.11a and IEEE 802.11p networking in ideal condition.

802.11a	**Time(sec)**	**802.11p**	**Time(sec)**
20 km/h	4.5	20 km/h	38.5
40 km/h	0	40 km/h	19
60 km/h	0	60 km/h	14

Table 2.4: 802.11p and 802.11a connection duration [66]

In contrast, forming a Service Set (SS) requires authentication, association, and time and frequency synchronization. The challenge in taking these steps is that they require a time interval that may not be suitable for some safety applications. During traffic movement, two cars may have access to the wireless connection in less than a second. To reduce the message latency, Y. J. Li [65] introduced a mode called Outside the Context of Basic Service Set (OCB). The OCB mode can be used in any two or more receiving devices within the area of a single connection. A mobile STA in the OCB mode can exchange data and control frames any time without forming or being a member of any SS. While minimizing latency, the OCB does not receive the authentication, association or data confidentiality services at the MAC layer. These services are partly transferred to the higher layer in IEEE P1609.2.

The network topology in IEEE 802.11p is a loose form of Independent Basic Service Set (IBSS), now called a WBSS, and this will not require authentication and association before joining a device. In a regular IEEE 802.11 network, an AP is responsible for sending beacons, and thereby, the synchronization of the network. In an IEEE 802.11 network using IBSS, this beaconing is distributed among the nodes in the IBSS cluster. In an IEEE 802.11p network, however, no beaconing exists, and the network synchronization depends on a global time reference, such as the Universal Coordinated Time (UTC), which allows synchronization and data exchange between two devices [57][67]. This can be provided by GPS. It is used to receive the UTC information through a management frame sent on the network.

2.5 Strength and Weakness of DSRC

Every communication protocol comes with a set of strengths and weaknesses; so also does DSRC. Many factors contribute to this state of affairs. Most of the major ones [7][68][69] are mentioned in the accompanying Table 2.5.

Strength	Weakness
Network is self-organizing and self-configuring. No authentication and association procedure at MAC layer. Low latency and high throughput. Low upfront cost. Effective, efficient and reliable for wide variety safety applications. Easy maintenance when in *ad hoc* mode. Wire equivalent security (WEP). Negligible signal interference from other bands. NLOS RF coverage < 1000m radius through single AP/OBU transceiver. Data transfer rate maximum up to 27 Mbps at 10 MHz channel. Support vehicular mobility at the speed of ~100 mph. Can operate on both a*d hoc* and Infrastructure mode for V2X communication. Dedicated safety channel known as CCH to seamlessly run safety critical applications. Availability of 6 service channels SCH for both safety and non-safety use. Availability of QoS support based on IEEE 802.11e. No constraints to energy for both OBU and AP. No subscription cost.	V2I communication, requires many AP's to cover entire city. Difficult to achieve seamless roaming in V2I communication because of non-availability of seamless horizontal and vertical channel handoff. Not suitable for applications that require continuous session because session breaks during Extended Service Set (ESS) transition. Rapid changes in topology causes frequent disconnections. Difficult to seamlessly deploy TCP/UDP based non-safety applications that co-exist with integral beacon based periodic safety messages. It does not scale to very large number of nodes. At higher vehicular density, the wireless medium becomes highly congested that causes collisions among packets. No centralized control to monitor devices on the network. Vehicles and stationary objects in a big number create multi-path fading. Weak connectivity in sparse vehicular environments. High vehicular speed can lead to multiple reflected paths in the channel. Sparse freeway environments are likely to generate poor packet delivery fraction.

Table 2.5: Strength and weakness of DSRC

2.6 Overview of TCP

The DSRC includes the widely-accepted transport layer protocol TCP to serve non-safety related applications. It is the most commonly used end-to-end, connection-oriented, reliable Internet Protocol to date, which ensures data delivery between a pair of nodes. It is said to be reliable because it efficiently implements flow and congestion control mechanisms. Apart from establishing communication between two nodes, TCP in general treats fairly with other transport layer agent like UDP without burdening the network.

TCP transmits data in the form of segments, limited to a Maximum Segment Size (MSS). During initial communication establishment phase, this MSS is fixed by mutual agreement between two communicating nodes through a three-way handshaking mechanism. Each transmitted byte of data is associated with a sequence number. When a node receives a segment, it notes down the sequence number range of the segment. For each received segment, the receiving node replies back with cumulative ACK, which assures the sender that the bytes up to the given sequence number has successfully been received. The TCP sender waits for some period of time for an ACK of the segment that has been sent out; on expiry of this time, an error is flagged by a timer, signalling that something has gone wrong. To implement this, TCP sender is provided with Re-Transmission Timeout (RTO) timer. Upon expiration of RTO timer, it is presumed that the segment has been lost and it is required to be re-transmitted.

At the TCP sender side, the bytes sending rate is restricted by the congestion window (*cwnd*), which limits the number of packets in transit, and slow-start threshold *(ssthresh),* which is the boundary between slow-start and congestion avoidance phases. The *cwnd* is tuned by the TCP sender according to different network conditions, and most importantly, it prevents congestion in the network. The receiving window (*rwnd*) is advertised by the receiving node, and is adjusted at regular intervals so that incoming segment does not exceed the receiving capacity; but the TCP sender controls and limits the data sending rate by using both *cwnd* and advertised *rwnd*. The size of the sending window (W) is defined as the minimum value of *cwnd* and *rwnd*.

$$W = \min(cwnd, rwnd) \tag{1}$$

TCP detects packet drop either when three consecutive duplicate ACKs are received, which is faster and efficient, or when RTO expires, which is slower and less effective. Transport layer protocol TCP is a window based flow and error control (congestion control) mechanism. It efficiently uses sliding window protocol to manage data transmission over network. The TCP sending rate is governed by two mechanisms: Flow Control and Congestion Control [70].

Flow Control: TCP-Flow Control mechanism controls the transmission rate at the sender side, so that it does not exceed receiver’s accepting rate. In general, devices cannot send and receive data at the same pace. If there is no flow control, these differences may lead to bandwidth wastage due to high percentage of packet drop. So, the flow control is used to prevent TCP sender from pumping data beyond the receiver’s buffer size. To achieve this, the receiver, when replying ACK, advertises receiver window limit to the sender using *rwnd* variable, which is basically the socket size of the receiver buffer available for the connection.

Congestion Control: When a source node sends TCP data faster than what a network can handle, congestion situation develops. The purpose of congestion control mechanism is to handle data traffic inside the network so that it can prevent such congestion. For this purpose, a variable, *cwnd* is issued by TCP sender to control the data packet sending rate so that pumped data do not exceed the capacity of network; otherwise network melt-down situation could arise.

2.6.1 TCP’s Congestion Control Mechanisms

As per RFC-2581 [71], TCP undergoes four congestion control phases *viz. slow start, congestion avoidance, fast retransmit and fast recovery*.

Slow Start: It is the initial phase of the TCP congestion control mechanism specified by RFC-5681. Upon establishment of an end-to-end connection, the slow-start begins with window size of one segment, which is MSS, initialized by the receiver. In the starting phase of slow-start, the sender first sends one segment of data and waits for its ACK from the receiver. After receiving ACK of the sent packet, the size of the *cwnd* gets doubled or increased by one segment for each ACK. In other words, the growth of the *cwnd* is exponential until the loss of first packet. The slow-start takes a long time during start-up phase, which is not good for high speed congested network, and hence it is not suitable for VANET. Besides this, if small data packets need to be sent frequently, then bandwidth efficiency will suffer greatly. In actuality, slows-start is not very slow when the network is not congested because upon successful transmission the *cwnd* increases exponentially.

Congestion Avoidance: Congestion window exponentially increases during slow-start phase. This exponential growth will continue to be in effect until either packet drop is detected or when receiver limits its *rwnd* or when slow-start threshold (*ssthresh*) is reached. If a packet drop event occurs, the TCP sender considers it as a congestion situation in the network, and as a result, reduces the offered load. Upon reaching *ssthresh,* TCP shifts from exponential to additive growth, where *cwnd* is increased by 1 segment for each Round Trip Time (RTT).

Fast Re-transmit: It is another important phase of TCP congestion control mechanism, which reduces TCP sender’s waiting time before re-transmission of the lost segment. In the event when RTO does not expire and TCP host receives three consecutive duplicate ACKs, then segment loss is assumed. In fast re-transmit phase, TCP host re-transmits the lost segment without waiting for the expiry of RTO.

Fast Recovery: Immediately after the occurrence of fast retransmission, fast recovery phase starts. During fast recovery, the *ssthresh is* adjusted to half of the latest window size, and because of receiving three duplicate ACKs, the *cwnd* size is adjusted to *ssthresh+3* segments. Usually Fast Re-transmit and Fast Recovery algorithms are implemented together.

Retransmission Time Out (RTO): It is a timer associated with each sent segment. RTO is maintained by the TCP sender in order to detect segment loss. Expiry of RTO timer is the indication of segment loss because of congestion or packet drop. TCP sender upon RTO expiry enters into slow-start phase, reduces the *cwnd* to one segment and re-transmits the lost packet.

Triple Duplicate Acknowledgments: It is another method adopted by TCP to detect segment loss. When a packet is dropped or delivered out of order at receiver side, it is considered that the packet-in-flight is lost due to congestion in the network. TCP receiver generates triple duplicate ACKs to inform the sender that there is packet loss.

In this case, the TCP receiver keeps sending ACK without modifying the Seq. No. field in the ACK packets. When the TCP sender notices receipt of three consecutive ACKs having the same packet sequence number, it concludes that the packet has been lost. During congestion avoidance phase, receipt of triple duplicate ACKs in a row causes the TCP sender to execute fast re-transmit and enter into fast recovery phase of congestion control mechanism. TCP sender upon receipt of all the successful ACKs adjusts the *cwnd* to *ssthresh*, stops fast recovery phase and restarts congestion avoidance phase anew.

2.7 TCP Variants

The fair treatment and efficient resource utilization achieved by TCP variants are not always explicitly mentioned in the literature [72]. There are many TCP congestion control variants that have been developed to monitor Internet traffic such as TCP-Reno [73], TCP-BIC [74], TCP-Scalable [75], TCP-Westwood [76], TCP-HighSpeed [77] and TCP-LP [78]. These TCP variants are perceived as most suitable and promising candidates for acceptance and efficient deployment in different VANET environments.

2.7.1 TCP-Reno

TCP-Reno [73] is the most widely used TCP congestion control mechanism implemented in the networks today. TCP-Reno follows all the algorithms which are described in TCP-Tahoe [RFC-2001]; in addition, it has Fast Recovery algorithm. With the use of Slow Start, Congestion Avoidance, Fast Re-transmit and Fast Recovery congestion control mechanisms, TCP-Reno efficiently handles the congestion in the network. It uses packet drops to calculate the available bandwidth in the network. After three-way handshaking, TCP begins with slow-start congestion control mechanism. The slow-start phase recurs when RTO expires at sender side during the communication.

In the beginning of the slow-start phase the *cwnd* exponentially increases. When the *cwnd* crosses *ssthresh* then with the additive increase the *cwnd* enters into Congestion Avoidance phase. After receiving three duplicate ACKs or before the expiry of RTO, the *cwnd* enters into Fast Retransmit and Fast Recovery phase. Fast Retransmit/ Fast Recovery mechanisms foster packet transfer speed with additive increment in the network because after receiving three duplicate ACKs, the *cwnd* does not start with slow-start phase.

After receiving three consecutive duplicate ACKs, the TCP sender, rather than going through slow-start phase, executes Fast Re-transmit and then enters into fast recovery phase. The TCP sender adjusts the *ssthresh* equivalent to half of the latest *cwnd and* the new *cwnd* equals to latest *ssthresh* plus three duplicate ACK. Each received duplicate ACK augments the *cwnd* size by one. When the TCP sender receives new ACK, it leaves the Fast Recovery phase and again enters into the Congestion Avoidance phase while resetting *cwnd=ssthresh.* In this way, the Fast Recovery mechanism maintains the average *cwnd* size high; as a result, performance in terms of achieved throughput is better than TCP-Tahoe.

For the single packet loss TCP-Reno works fine but if multiple losses occur from the same *cwnd*, the performance degrades extensively because it repeatedly exits and re-enters Fast Recovery phases or enters into timeout. TCP-Reno operates similar to TCP-Tahoe in case of timeout situation by adjusting the *cwnd* = 1 packet and again enters into slow-start phase.

The Additive Increase and Multiple Decrease (AIMD) mechanism of TCP-Reno increases the *cwnd* by one segment per RTT for each received ACK and *cwnd* reduces to half for each packet loss per RTT. The AIMD *cwnd* is adjusted by:

$$\text{Increase: } cwnd = cwnd + 1/\,cwnd \qquad (2)$$

$$\text{Decrease: } cwnd = cwnd - (1/2)\,cwnd \qquad (3)$$

2.7.2 TCP-BIC

TCP-BIC (Binary Increase Congestion Control) [74] is an optimized but complex congestion control mechanism targeting high speed network which ensures faster convergence and RTT fairness. TCP-BIC uses Additive Increase as well as unique Binary Search Increase algorithm that allows TCP to occupy available network bandwidth faster and retain the window for a maximum period of time.

When no packet loss occurs the Binary-Search algorithm adjusts the window size to minimum (*Wmin)* but the moment TCP encounters packet loss, it is immediately set to maximum *(Wmax)*. The optimum window size is located in the middle of *Wmax* and

Wmin. At regular intervals, the current window size is adjusted in between these two sizes and feedback is received in terms of packet loss.

The regular adjustment of the window size is made by finding the midpoint of the received feedback. If a packet loss occurred the midpoint of the feedback is set to *Wmax* or *Wmin* in case of no packet loss. TCP-BIC utilizes the network bandwidth efficiently by aggressively probing in slow-start phase for available bandwidth, whenever there is big difference between *Wmin* and *Wmax*. But as these differences reduce, the current window growth becomes less aggressive; as a result, one gets logarithmic decrement of packet loss. Linux kernel from 2.6.8 till 2.6.18 uses the TCP BIC as default TCP protocol but it changed to TCP-CUBIC in the version 2.6.19.

2.7.3 TCP-Scalable

TCP-Scalable (STCP) [75] is formulated with a small modification in the sender side TCP *cwnd* update mechanism. It dramatically improves TCP performance in high speed wide area networks using conventional TCP receivers. STCP behaviour is identical to traditional TCP stacks when small window sizes are sufficient and its design targets incremental deployment. The STCP and HighSpeed-TCP were primarily formulated to be used in high-speed guided links, and they seem to be the strongest candidates in the next generation Internet to replace the current congestion control mechanism implemented by standard TCP. STCP also uses Multiplicative Increase Multiplicative Decrease (MIMD) technique to achieve better utilization of a network link with high bandwidth delay product.

According to RFC-2581 and RFC-5681, the *cwnd* should be scaled down to 1/2 for each packet loss and this process will continue until packet drop stops. Once the packet loss stops, the slow-start phase starts with exponential increment of window. In STCP, instead of halving the *cwnd* for each packet loss, the *cwnd* reduces to 1/8 until packet loss ceases. Once the packet loss stops, the window is increased with fixed rate; one packet is added after the receipt of every 100 successful ACKs. The modified algorithm to update TCP's congestion window is given below:

$$\text{Increase: } cwnd = cwnd + (1/100)\, cwnd \tag{4}$$

$$\text{Decrease: } cwnd = cwnd - (1/8)\, cwnd \tag{5}$$

2.7.4 TCP-Westwood

TCP-Westwood (TCPW) [76] is sender side modification of TCP Reno algorithm which optimizes TCP congestion control performance over both guided and unguided media to handle large bandwidth delay product. However, this algorithm provides substantial improvement over lossy wireless network. TCPW uses bandwidth estimation that operates at TCP sender side in order to find whether the packet loss is from wireless channel error or congestion in the network. TCPW algorithm ceaselessly calculates the connection BandWidth Estimate (BWE) that is defined as the share of bottleneck bandwidth used by the established connection. Therefore, BWE is equal to the rate at which data is delivered to TCP receiver.

In the event of packet loss, TCP-Reno halves the *cwnd* size which decreases the TCP performance, whereas TCP-Westwood estimates the available bandwidth of the connection using rate of received ACK and on their payload. After the receipt of three consecutive duplicate ACKs or when timeout events occur, the sender resets the *cwnd* and *ssthresh* based on BWE. This method leads to faster recovery. This algorithm is especially effective over wireless channel where intermittent losses due to radio channel errors are usually misinterpreted as a cause of congestion by current TCP schemes which result in unnecessary window reduction. Setting of congestion window and slow-start threshold in TCP-Westwood go by the following scheme:

- After receiving 3 duplicate ACKs: ssthresh = (BWE * RTTmin)/Segment size
- If (cwnd > ssthresh) cwnd = ssthresh
- After timeout: ssthresh == (BWE * RTTmin)/Segment size

2.7.5 TCP-HighSpeed

TCP-HighSpeed (HSTCP) [77], formally defined in RFC-3649, applies minor changes to TCP-Reno congestion control mechanism for TCP connections that has very large congestion windows. TCP-Reno in high bandwidth link requires low packet loss rate to fully utilize bandwidth and it takes much time to achieve higher throughput. This is mainly due to use of AIMD congestion controls mechanism used by TCP-Reno that requires several RTTs to get enough *cwnd* size. HSTCP also follows AIMD mechanism but it swiftly increases and slowly decreases the *cwnd* in order to keep the *cwnd* space just enough to meet high bandwidth delay product requirements [79] In HSTCP the increase and decrease of *cwnd* in response to single acknowledgment and congestion event are as follows:

Increase: *cwnd* = *cwnd* + a (*cwnd*) / *cwnd* (6)

Decrease: *cwnd* = (1-b (*cwnd*)) *cwnd* (7)

For standard TCP, a (cwnd) =1 and b (cwnd) =1/2, regardless of the *cwnd* value. HighSpeed TCP uses the same values of a (cwnd) and b (cwnd). HSTCP and TCP-Reno are similar in terms of packet loss rate up to $10^{3.}$ The HSTCP response function (the function mapping the steady-state packet drop rate to TCP's average sending rate in packets per round-trip time) could be expressed by cwnd = $0.12/p^{0.835}$. For this reason, the *cwnd* is not increased by 1 packet for each ACK, but uses a dynamic variable that relies on current *cwnd* value in the congestion avoidance phase of HSTCP. Similarly, when packet loss event occurs the multiplicative decrease is dynamic too but it is lower than (1/2) *cwnd*.

The modified response function of HSTCP is effective with higher *cwnd* and it does not alter TCP behaviour in heavily congested environments, and therefore it does not change the behaviour towards congestion collapse. In order to modify response function HSTCP uses three parameters: Low_Window, High_Window, and High_P (P as packet drop rate). HSTCP accomplishes TCP compatibility by using the same response function as TCP-Reno when the current *cwnd* is at most Low_Window. Though HSTCP response function is applied when the current *cwnd* is greater than Low_Window, the property of HSTCP response functions give a straight line on a log-log scale similar to standard TCP response function for low to moderate congestion. If the value of average congestion window W is greater than Low_Window, then the response function becomes:

$$\mathrm{W} = \left(\frac{\mathrm{p}}{\mathrm{Low\_P}}\right)^{\mathrm{S}} \mathrm{Low\_Window} \tag{8}$$

Where Low_P is the packet drop rate and S is constant,

$$\mathrm{S} = \frac{(\log\ \mathrm{High\_Window} - \log \mathrm{Low\_Window})}{(\log \mathrm{High\_P} - \log \mathrm{Low\_P})} \tag{9}$$

The main purpose of modifying HSCTP is to occupy the available bandwidth more aggressively than TCP-Reno, which easily adapts the TCP window to high bandwidth link.

2.8 TCP in VANET

Though DSRC is specifically designed for time sensitive real-time safety applications, it has provision to also incorporate TCP at transport layer to serve delay-tolerant infotainment services. Most of the non-safety comfort applications are based on direct link with RSU and vehicles. The most common Internet protocol TCP is not designed and not suitable for unpredictable nature of wireless channel [77]; however it performs better in contrast to UDP. Its use over the MANET/VANET is a necessary requirement to provide infotainment data. Not only is TCP/IP the worldwide accepted standard network protocol stack on the Internet and influences a large number of applications, but it also allows seamless integration with the Internet, wherever available [81]. Downloading content such as music or road maps, using V2I network is considered to be a useful delay-insensitive application.

Contents are downloaded by using the existing Internet protocols, for instance, the File Transfer Protocol (FTP). The files are broken into many parts and are sent via TCP. For each part of file to be transferred by TCP, two packets are generated by the sender and the receiver, one for TCP data segment and another for TCP acknowledgment [33]. The performance of both TCP and UDP can be measured with regard to non-safety applications over DSRC. TCP may be used for any downloading service while UDP is suitable for bursty as well as non-bursty applications (such as VoIP or Video calls). In 802.11a, TCP performs very poorly in terms of throughput, the reason behind this being the presence of both non-safety UDP applications and safety non-TCP-friendly applications. By contrast, DSRC shows significant improvement in the performance of TCP, as there are dedicated time slots for each type of applications [82].

The basic properties of wireless radio communications are fundamentally different from wired networks. Several investigations on the impact of these properties on the performance of TCP show that TCP provides poor throughput in dynamic wireless network [80][83]–[86]. This poor performance is solely due to the conservative flow and congestion control mechanisms followed by TCP [87]. In wireless networks TCP interprets transmission errors as a congestion situation; as a result, it reduces the consumed bandwidth [88]. Though congestion is not the primary cause of the packet losses in wireless network, it may arise due to the high Bit Error Rate (BER) and frequent connection disruptions.

High node mobility in VANET results in frequent network disconnection and re-connection events, which adversely impact TCP performance [89]. The traditional TCP cannot guarantee reliability and high efficiency in transmitting data over dynamic wireless medium, because it is not designed to take into account the wireless network characteristics, such as multi-path signal propagation, limited bandwidth, BER, long latency and unexpected link breakage.

The reliability of TCP is assured through the efficient implementation of flow and error control. TCP flow control mechanism controls the transmission rate at the sender side, so that it does not overwhelm receiver’s accepting rate. In general, devices cannot send/receive data at the same speed. If there is no flow control, these differences may lead to bandwidth wastage due to high volume of packet loss. TCP congestion control is the mechanism to handle congestion situation to prevent network melt-down. Traditional congestion control mechanism of TCP does not perform well in vehicular environments. This is because the predictions about potential congestion situations are based on local information, which may not reflect the current state of the network [90]. Developing effective congestion control mechanisms is also a very challenging task for VANET environment.

TCP uses window and acknowledgement based system to control the flow of packet. The transmission window starts with exponential increase and additive increase when three duplicate acknowledgments are received or timeout has reached. The issue with highly dynamic topological VANET is that the transmission medium is not reliable; as a result, the information is subject to bit errors which may completely damage it. When the packets are disrupted, the *cwnd* shrinks, thereby reducing throughput [90]. In this situation, the disrupted packet needs to be resent and kept sending at the maximum transmission rate. But, TCP slow-start congestion control mechanism starts with very low transmission rate, which is increased with the reception of every acknowledgement.

Furthermore, the slow-start congestion control mechanism reduces the *cwnd* size and increases the re-transmission timeout value by two (an exponential back-off) [70]. This slow-start process prevents full utilization of bandwidth till the entire session. Such unnecessary congestion control decreases the effective utilization of bandwidth, and as a result, significantly reduces the performance. In addition, a succession of timeouts at the TCP sender side lowers general throughput more than the packet losses do. Serial timeout that is caused by frequent disconnection is a failure of the TCP sender in not receiving acknowledgement constantly. Thus, effective and efficient congestion control mechanism is needed to maintain high reliability of the infotainment services in VANET environment, while providing fair share of bandwidth to crucial safety applications.

2.9 Related Work

In this section, several publications and journal articles are reviewed in which various authors have concentrated on Modulation Scheme, Safety Message broadcast generation/interval rate and TCP congestion control over VANET. A comprehensive yet concise survey of which follows.

2.9.1 Related Work on Message Interval Rate and Modulation Scheme over VANET

The IEEE 802.11p standard does not specify the optimum periodic safety messaging generation rate and modulation scheme, but in the literature [91]–[96], one finds that the typical message frequency ranges from 0.05s to 0.1s may be used. However, message generation or message interval rate and modulation depend upon types of cooperative awareness safety applications required to be deployed and the traffic condition. When two vehicles are in close proximity, the probability of collision obviously increases. So, the higher the sensitivity of safety application or traffic situation, the more the packet generation per second with strict low latency is needed in order to prevent accidents. These periodic safety messages in VANETs are commonly known as BSM in the United States and Cooperative Awareness Message (CAM) in Europe [97]. Here, in this research Periodic Broadcast (PBC) safety messaging application agent is used which is designed specifically for VANET, whose detailed description is given in Chapter 3.

Several previous works studied on adaptive modulation schemes over VANET, but here in this work, performance analysis of 802.11p defined singular modulation scheme with respect to density of vehicles is made. It is clear from previous research that low index modulation schemes like Binary Phase Shift Keying (BPSK) & Quadrature Phase Shift Keying (QPSK) having low data rate exhibited low BER over wider communication range with low Signal to Noise Ratio (SNR). However high index modulation scheme 64-QAM, proved to be good for higher remission rate with high SNR, but closer distance between nodes. In the past few years, VANET research community has made considerable effort on improving time-sensitive safety applications as can be seen in [28][96][98]-[107].

Christopher et al [28] focused on adaptive modulation to improve the data throughput and efficiency of channel spectrum in VANET environment. Through theoretical and simulation studies, it was shown that adaptive modulation performs well and improves throughput better than other modulation schemes, when used singularly in highly dynamic VANET. These authors examined and presented adaptive modulation in conjunction with speed of the vehicles, considering non-safety UDP traffic over 802.11p based VANET. They found 16-QAM and 64-QAM scheme perform better if the distance between the vehicles is close. So, when the SNR is high they provide higher bandwidth whereas BPSK and QPSK provide low data rates but can cover a wider range with low SNR. The authors have considered UDP of 512 kb as the main traffic for the analysis; however, in real VANET scenario, there should be integral beacon based periodic broadcast of short messages (100-300 bytes) as the main traffic to gauge the protocol performance. Also, these authors have not mentioned traffic environment and experimented with mobility simulator to precisely estimate the vehicular speed over which channel condition can be assumed.

Schmidt et al [96] presented a detailed analysis of how different beacon rates influence the offered load to the channel and the resulting average and maximum accuracy of information. They proposed schemes for adapting the beacon rate according to the traffic situation.

Chang et al [98] have proposed an efficient, on-demand adaptive channel estimation technique based on prediction of the SNR to provide consistent bandwidth with QoS in VANET highway environment. The estimator is based on the Received Signal Strength (RSS), deployed over UDP data traffic, to provide required feedback to the node engaged in communication, without modifying the existing protocols. Hassan et al [99] proposed joint adaptive modulation and channel coding scheme for image and video over fading channels. In that work, the modulation index and channel coding rates are elected based on the sensitivity of the compressed bit stream, while taking into consideration the wireless channel conditions to meet a predefined target BER. Lye et al [100] compared the performance of wireless communication with adaptive modulation and coding in VANET against rigid transmission technique. These authors demonstrated with simulation study that in comparison to fixed QPSK modulation the adaptive modulation outperforms.

Javed et al [101] presented a combined transmission range and packet generation rate control algorithm which takes into account the safety of the vehicles and maximizes the control channel CCH utilization. They have examined adaptive BSM packet generation rate rather than fixed periodic broadcast with respect to transmission range as well as density of the vehicles, while using 6 Mbps data rate. The performance analysis showed that the proposed algorithm improved the safety message performance in terms of packet reception rate and control channel utilization for a range of vehicle densities and vehicle speeds. Further, their result showed that the proposed algorithm performance was much better than the fixed transmission range and packet generation rate scheme with respect to number of vehicles. These authors have studied the effect of fixed 10 packets per second generation rate compared with adaptive one. They have, however, not demonstrated and compared the effect of variable packet generation rate with respect to vehicle density.

Park and Kim [102] describe a light weight application-level messaging generation estimation scheme called Frequency Adjustment with Random Epochs (FARE), which significantly improves the BSM throughput, while using less bandwidth than what 802.11p delivered. They have also demonstrated that BSM application in the WAVE communication environment can control the messaging generation rate in the absence of either feedback information from explicit communication or cross-layer assistance specifically from the MAC layer. Further, they have shown through simulation and analysis that by simply imposing an application timing structure over BSM message transmissions, FARE can enable the BSM application to estimate the vehicular traffic density in the proximity and accordingly regulate the messaging frequency. The authors have not studied and compared the effect of variable packet generation rate with respect to vehicular density. Moreover, the mobility simulator to gauge the impact of channel contention and speed of the vehicles in a realistic highway like scenario is not mentioned in their simulation setting.

Huang et al [103] proposed a joint rate & power control algorithm for broadcasting of self-information that enables neighbour tracking in VANET. The proposed solution used a closed-loop control concept and accounts for wireless channel unreliability. They evaluated the proposed solution through realistic network and microscopic traffic simulations, whose results confirm that if packet generation rate and associated transmission power for safety messages are adjusted in an on-demand and adaptive basis, then robust tracking is possible under various traffic conditions. The results confirm that the proposed design is robust and can considerably reduce the tracking error compared to that of the *de facto* solution proposed by VCC i.e. beaconing at 100ms or 500ms intervals. These authors have not compared tracking with beaconing using the most important metric end-to-end delay to evaluate the safety messages performance.

Rahman and Nasiruddin [104] using NS2.35 studied the impact of IDM-IM and IDM-LC mobility model for vehicular safety applications. Using 9 different metrics the authors theoretically discussed PBC traffic but considered only CBR traffic in their simulation. In addition, they considered fixed packet generation rate of only 4 packets per second in their scenario. The authors did not mention reasons for not using PBC agent, which is specifically designed in NS2 to simulate VANET safety applications. Yousefi et al [105] evaluate the performance of a single-hop dissemination protocol, while taking into account the QoS. Further, the effects of three parameters *viz.* vehicle's transmission range, message transmission's interval time and message payload size are studied. The authors have considered CBR traffic over only two transmission rates (100 & 200ms). The CBR traffic can be assumed as safety beacon like traffic but in the scenario it is not mentioned it is behaving like standardized PBC traffic.

Jafari et al [106] analysed and evaluated the effect of varying vehicle speed and different message sizes in DSRC based VANET using NS2 network simulator. This work has considered three lane highway traffic scenario, having only a total of 10 vehicles at freeway traffic scenario. The authors have not mentioned minimum simulation parameter details. Apart from these, when using PBC agent, the payload size of BSM exceeding 300 bytes is not the recommended safety short messaging size as per ITS.

Yang et al [107] proposed channel adaptive broadcasting method, which depends upon channel condition information available at each vehicle by employing standard supported sequence number mechanisms. The proposed method is fully compatible with IEEE 802.11 based networks and introduces no communication overhead. Simulation studies showed that it performs better than standard broadcasting in term of reception rate and channel utilization. They showed by means of analytical modelling and simulation study that by adaptively adjusting the transmit power, the proposed one-hop broadcasting protocol increases the reception rate at closer distances and alleviates collision possibility at further distances. Their work achieves performance similar to existing work, while having the merit of standard compatibility and low overhead. They have considered fixed message generation rate of 10 messages per second over only BPSK modulation scheme. The simulation study has not considered vehicular velocity and network environment.

Currently, there is no final recommendation from IEEE for a particular periodic broadcast rate or message interval rate for safety short messaging applications. Further, no requirements for the diverse safety applications have yet been explicitly defined. Most of the above proposed solutions aim to select the adaptive transmission rate and adaptive modulation scheme in order to improve the throughput and reception probability of the BSM. No work so far has dealt with optimum realistic BSM broadcast interval rate with respect to density of vehicles as well as analysed optimum modulation scheme for beacon based safety messages.

2.9.2 Related Work on Use of TCP over VANET

TCP was originally designed for wired network; yet, it gives acceptable performance in static wireless network like Wi-Fi. The highly dynamic nature of VANET is indicated by the high variation in transmission quality that greatly differs from Wi-Fi like wireless communication. TCP kind of applications are hard to make available continuously and seamlessly because of frequent disconnection in VANET environment. There are various studies that have investigated these aspects on the performance of TCP. Previous investigations showed that TCP is unable to provide good throughput in multi-hop *ad hoc* networks; however, theory points to higher possible throughput [108]. The poor performance is mainly due to TCP's conservative flow and congestion control mechanisms. For instance, TCP interprets transmission errors as a congestion situation by reducing throughput as a result. To mitigate this, TCP slow-start and congestion avoidance algorithms are pressed into service.

TCP was enhanced with many new features after the launch of its original version. According to RFC-2582, the TCP-Reno proposed fast-retransmit and fast-recovery congestion control algorithm, which was further refined in TCP NewReno. As per RFC-2018, TCP was further enhanced with Selective Acknowledgement (SACK). Later on, several improvements were made to congestion control algorithm by adding unique features to base TCP such as BIC, CUBIC, Scalable, Highspeed, Hybla and LP. All these enhancements can be broadly categorized into three classes; Loss-Based, Delay-Based and Loss-Delay based. Some of them are integrated into UNIX, Linux and other OS kernels in their TCP implementation. However, such existing extensions do not address the crux of the problem raised by TCP in VANET environment.

The majority of conventional TCP evaluation studies over VANETs have been carried out with simulations. The unsatisfactory performance of TCP over lossy vehicular wireless communication makes application deployment difficult, especially when non-TCP traffic tightly occupies the bandwidth. There are several works on TCP or TCP variants over MANET. Due to high mobility and harsh fading environment, the performance of MANET protocol yields anomalous results in VANET scenario. For these reasons literature survey on TCP over MANET are not reviewed. There exists a few studies [88] [109]-[121] that have examined the use of conventional TCP or TCP variants in VANET environment.

Chen et al [88], analysed the performance of multi-hop TCP in a multi-lane freeway environment, where vehicles are configured as clients or routers trying to reach a fixed AP. These authors stressed on the effect of tuning transmission power in dense and sparse road condition on throughput and latency of the conventional TCP. Using tailored simulation tool, they obtained detailed TCP statistics, and correlated them with the location, velocity and other properties of VANET nodes. In all defined case scenarios, the results obtained showed that the throughput significantly deteriorates as the number of hops increases from the sender to the AP. The higher the number of hops, the higher the losses due to interference and loss of connectivity, which causes TCP to throttle back the sending rate unnecessarily.

Spaho et al [109] investigated TCP traffic in VANET. They investigated and compared using good-put metric under DYMO routing protocol for TCP-Vegas and TCP-NewReno congestion control algorithms. The simulation results showed that TCP-NewReno performs better than TCP-Vegas in VANET scenario. Their experiment showed that TCP-NewReno uses larger *cwnd* compared with TCP-Vegas, which indicates better utilization of available bandwidth. TCP-Vegas could not differentiate congestion from packet losses due to link failures, so it reduces the *cwnd* size resulting in reduced good-put. However, TCP-NewReno over DYMO offers better good-put compared with TCP-Vegas. Also, TCP-NewReno exhibited faster recovery compared with TCP-Vegas and it performed better in VANETs scenario. However, they have not considered DSRC media access protocol, vehicular speed and traffic other than TCP on the channel in their experimental investigations.

Spaho et al [110] in their next experiment examined the performance of OLSR and *Ad hoc* On Demand Distant Vector (AODV) routing protocols when sending triple TCP-NewReno flow traffic over FTP application in VANET environment. Using DSRC communication protocol and TwoRayGround radio propagation model, these authors considered two data rates of 0.1 Mbps and 1 Mbps. The simulation results showed that, for smaller data rates, routing had huge effect on the network performance wherein AODV exhibited better link stability and higher throughput when compared to OLSR. Both the protocols showed almost same performance for higher data rates. This is because congestion, node stack overflow and routing played only a small role on dynamic network performance. Though these authors mainly evaluated the performance of routing protocol considering TCP traffic in perfect VANET scenario, realistic vehicular wireless network requires one to consider integral PBC like safety messages as the main background traffic. Only then can a genuine VANET environment be realized, enabling a better analysis and evaluation of the performance of each layer of protocol.

Pirmohammadi et al [111] proposed MAC sub-layer based approach to improve fairness in TCP and UDP flows coexisting in VANET. They have used both TCP and UDP traffic simultaneously in the scenario but not considered DSRC - MAC & PHY layer protocols as well as real vehicular network environment (Urban, Rural or Highway) using mobility simulator. The assumption of static vehicular scenario considered in that work does not make any difference with Wi-Fi connection between nodes in a home environment. Realistic VANET scenario at least requires periodic broadcast of safety messages as integral traffic by moving nodes at an average speed of 40-60 km/h. This is because the vehicular speed, integral beacon traffic and node density interact in a complex way impacting the wireless channel, which in turn affect PHY-MAC layers, with the problem then eventually getting escalated to upper layers. These issues have not received due consideration in their experiment.

Rahim et al [112] analysed the performance of TCP and UDP traffic in VANETs by using 802.11e and compared it with 802.11. They have considered both TCP and UDP simultaneous traffic in the scenario but not used DSRC protocol as well as real vehicular network environment (Urban, Rural or Highway) using mobility simulator. Here, in both these cases, they have considered TCP for high priority emergency data, which does not reflect the real VANET scenario. ITS inevitably requires emergency messages to be broadcast, not unicast; however, TCP based applications mostly use unicast services for non-safety comfort applications. They proved through simulation that performance of TCP traffic in VANETs increases when using 802.11e rather than 802.11p protocols.

Jun-Li Kuo et al [113] analysed the performance of real-time streaming under TCP and UDP in VANET. These authors propose a V2V streaming to build a V2V evaluation of real-time multimedia streaming under TCP/UDP on VANET. They compared TCP with UDP under real-time multimedia delivery. TCP usually has high fairness and high continuity, and UDP usually has high throughput in the wireless network. Integral background PBC traffic has not been accounted in the comparison which is an essential requirement in VANET based on DSRC. Even for 802.11 based protocols other than DSRC, there must be default periodic safety message broadcasting in VANET environment in order to fulfil the main purpose of VANET communication. In addition to this, they have assumed TCP congestion phenomena and timeout as in normal mobility scenario. However, it is well known from previous studies that TCP congestion control hugely impacts on TCP performance in VANETs.

Paul et al [114] compared AODV, Dynamic Source Routing (DSR) and Destination Sequenced Distance Vector (DSDV) routing protocols with varying pause time and node density over TCP and CBR traffic in VANET environment. The performance of reactive (AODV, DSR) and proactive (DSDV) routing protocol has been analysed by means of Packet Delivery Ratio (PDR), packet drop ratio & average E2ED performance metric, with varying pause time and node density. Though the authors mainly evaluated these routing protocols in their study, we can estimate the performance of TCP and CBR traffic, as these are the main data agents in their simulation. They concluded with their experiment that for sparse node with low pause time, the PDR of CBR connection for these routing protocols is low but for TCP traffic it is high in DSR and average for DSDV. Since CBR traffic is very similar to PBC safety messaging, which is the most important data traffic over VANET, the authors did not investigate what if both CBR and TCP traffic co-existed over single channel, so that the performance of routing protocol could have been analysed in realistic VANET scenario. Also, their experiment did not mention the communication protocol and propagation model used. Presumably, MAC and PHY of 802.11 might have been used in the experiment. A VANET over 802.11a greatly differs in performance from 802.11p communication protocol [58].

Jude and Ganesan [115] made comprehensive experimental analysis on TCP congestion control variants in vehicular wireless network. These authors considered loss-based, delay-based and loss-delay based congestion control mechanism over 802.11p based VANET. According to these authors, the TCP performance, in terms of throughput, delay and packet drop, severely deteriorates in VANET environment. The throughput showed less than 1.5 Mbps with the utilization rate of 6% for higher number of flows. Moreover, sharp increase of delay and maximum number of packet drop were observed for 150 flows. Various parameters were undefined in their experiment, such as modulation scheme using 802.11p protocol over which data rate depends, application type over TCP and speed of the vehicles. It is also not mentioned whether they have used p1609.4 multi-channel operation or TCP flowed over single channel. Realistic vehicular wireless experiment is required to have mobility model that defines the VANET environment condition, including micro or macro vehicular mobility. Apart from all these, the simulation scenario must consider periodic safety messages that should co-exist with TCP flow. Then only real objective of 802.11p protocol based VANET can be realized.

Jaiswal and Bhadauria [116] made performance analysis of UDP/CBR & TCP/FTP traffic under reactive and proactive routing protocols in urban VANET scenario with varying node densities. Taking into consideration packet delivery ratio, end-to-end delay and throughput metric, they found that FTP-TCP offers a far better performance than CBR-UDP in terms of packet delivery ratio and throughput for all routing protocols, while the performance of CBR-UDP traffic is better in terms of end-to-end delay for all routing protocols. The experiment has not considered high priority safety messaging traffic with TCP/UDP which is necessary to emulate real VANET like scenario.

Kaur and Josan [117] investigated the performance of TCP congestion control variants in VANET environment. They considered delay and throughput metrics to analyse TCP variants and drew conclusions based on the evaluation results using OMNET++ and SUMO simulator. They showed that performance in terms of throughput of TCP-NewReno is better than TCP-Reno, but the performance of Tahoe achieved less delay and better throughput in a large sized network. The work has not mentioned any simulation parameter including communication protocol used, and most importantly, they have considered only three vehicles in the scenario.

Dalal et al [118] investigated the performance of TCP and UDP in different VANET environments. They used throughput, number of packet drops and number of collision packets as performance evaluations metrics. The experiment was conducted with 802.11b protocol in *ad hoc* mode. They observed higher UDP throughput, while collision and packet drop rate of TCP are considerably less. They have not considered UDP as a high priority safety data traffic; rather a separate experiment was carried out. While using TCP over VANET, one must consider beacon like integral periodic safety messaging traffic in the scenario.

Al-Hasanat et al [119] introduced two enhancements in TCP-Westwood i.e. fast re-transmission and fast recovery procedures over high bit error network. The modification is introduced to rapidly activate fast re-transmission mechanism by preventing the TCP sender from waiting for the third duplicate ACKs to re-transmit lost packets. Their second modification achieves better recovery for the *cwnd* size in the fast recovery phase based on last RTT and bandwidth estimation. The authors compared the throughput and *cwnd* size of the new proposed modifications with those of TCPW and NewReno. The outcome of their proposed algorithm showed significant improvement over the two implementations. The experiment is not conducted over dynamic topological network (MANET/VANET) where lossy channel condition affects the performance of MAC layer and the layers above.

Henna [120] made throughput analysis of TCP variants in mobile wireless network. This author has compared the performance of several TCP variants under random packet loss rate, and also analysed the impact of mobility on the combinations of TCP-New Jersey and TCP-Vegas, with three frequently used routing protocols such as DSR, AODV and DSDV. She concluded that the performance of all three routing protocols decreases when mobility increases, and TCP-New Jersey is preferable under very high Random Packet Loss Rate. She has considered mobility up to 70 km/h, which is not typical of MANET, but rather of VANET. Also, the author has not considered periodic broadcast traffic along with TCP and did not consider the impact of network environment upon protocol.

Hadrien and Cottrell [121] compared and evaluated the performance of TCP-NewReno with HSTCP, Fast TCP, S-TCP, HSTCP-LP, H-TCP, and BIC-TCP on high-speed production networks. Their experiments indicate that TCP-NewReno showed poor and unstable performance, whereas most of the high-speed protocols delivered performance improvement over TCP-Reno. These experiments did not have control over background traffic. They used UDP as background safety data traffic and did not consider the impact of network environments.

The primary purpose of VANET is to support safety applications that rely on periodic broadcast of messages to prevent accidents. If we try to use TCP over VANET, it should be done in a way that does not disturb or hurt integral beacon based time sensitive data traffic. Most of the research work on non-safety application over VANET have considered TCP as singular data traffic without including, in parallel, integral broadcast of safety data flow in their scenario. And, a few of them have considered non-broadcast based CBR as the safety data traffic while using TCP connection. In view of this, such a study cannot be taken as a realistic VANET simulation work.

The performance of use cases which depends upon TCP agent may severely suffer because of frequent disconnection and channel handoff. Successive channel disruption forces the moving node to re-start TCP connection with the roadside infostation from slow-start phase, and as a result the performance of non-safety application suffers. Also, the aggressive nature of slow-start algorithm makes the TCP application grab the bandwidth greedily, thereby bringing down the performance of crucial safety application.

To overcome the under-performance of TCP in dynamic topological network, many substitutes for TCP have been proposed, such as SCTP [25], ETP [26], CTP [27], MOCCA [80] and MCTP [87], and all of them do perform well. However, the problem of adopting and deploying these proprietary protocols is backward compatibility as well as suitability for diverse applications and platforms that are yet to be verified.

It has been more than three decades since the TCP has been in use and thousands of applications have been designed, developed and deployed on the conventional architecture of TCP/IP. Replacing TCP with a suitable substitute could cause severe disruption in service and might raise many functional issues with the applications which are already deployed. So, the best possible interim solution - and indeed the most pragmatic way to go about - is to take the course of modifying and optimizing the existing conventional TCP, without losing sight of the dynamic network condition and more importantly, the requirements of BSM broadcast.

2.10 Chapter Summary

After a brief look at the MANET and VANET technologies, this chapter has described the architecture of DSRC within the framework of VANET, explaining the structure and function of relevant layers. Also described were its networking services, and authentication and association processes, following which the strengths and weaknesses of DSRC were highlighted. Next, the traditional TCP and its congestion control mechanisms were examined critically. Many variants of these algorithms designed over the years to make the standard TCP work efficiently in VANET environment were reviewed.

Previous work on employing different message interval rate and the suitable modulation schemes were surveyed, and their inadequacies or lacunae in the present context of DSRC over VANET were pointed out. Then the literature on using TCP variants in dynamic environment were presented. It is found that heterogeneous reliable transport layer traffic that can co-exist with safety application over WAVE architecture has not been evaluated in a realistic VANET environment. Also, the original architecture of TCP had not been designed foreseeing the later day technological advances and so it requires suitable amendments to cope with DSRC requirements in a VANET environment. This becomes all the more necessary in the face of many interlocking parameters (like modulation scheme, broadcast interval rate, vehicle density on the road, congestion control variants etc.) interacting in a complex way that influence (mutually) the performance of delay-sensitive safety messages on the one hand, and delay-tolerant TCP applications on the other, when they coexist.

Therefore, the central research question that presents itself now – indeed the principal objective of this study – is how to make the TCP based applications get along with the delay-sensitive safety data traffic in as smooth a manner as possible and with as little interference as permissible. This issue features as a recurring theme running throughout the thesis; it is indeed the *leitmotiv* of this work. By bringing into prominence this *uneasy* relationship or *tension* arising between core safety related applications and the TCP-based infotainment applications, this research work focuses on this pressing problem in an effort to bring about some kind of resolution. The solution arrived at consists in devising an innovative method to adaptively modify and regulate the behaviour of congestion control mechanism, after identifying the suitable modulation scheme and an appropriate BSM broadcast interval rate.

The chapters to follow present in considerable detail a succession of simulation activities, logically related and functionally required, that are carried out both to articulate the necessity or desirability of the proposed, newly modified congestion control mechanism and to investigate the implications of implementing it in a VANET environment. Closely woven into this account, these chapters also contain discussion that examine how far the new proposal is effective in addressing the core concerns of our research agenda and in particular, how successful or satisfactory it has been in realizing the principal objective of this dissertation.

Chapter 3

VANET Simulation, Performance Metrics and Experimental Scenarios

In this chapter, various considerations that go into setting up a comprehensive and successful VANET simulation are presented. Beginning with traffic environment which is the starting point for VANET deployment, the importance of an appropriate mobility model is mentioned and so also the usefulness of MobiSim tool to generate freeway mobility model. Then follows the details of the most powerful network simulator NS2, which forms the backbone for all the simulation studies conducted in this research, and the way MobiSim feeds and interacts with NS2. Routing protocols are very crucial in deciding the performance of protocol and some of the major ones specially designed for freeway mobility model in a VANET scenario are then described. Other related prerequisites for DSRC based VANET simulation like propagation model, BSM data traffic source and modulation schemes also find their due place in this chapter. General performance analysis metrics which are normally used to study and evaluate VANET simulation are defined and described. Finally, two simulation scenarios are constructed that form the focus for studying the performance of both BSM application and TCP based non-safety application.

3.1 Traffic Environment for VANET simulation

To simulate real vehicular wireless network and traffic, it is essential to consider significant characteristics and features of surrounding mobility environment that truly have a bearing on protocol performance. Simulation approach that allows analysing factors impacting throughput under different environmental conditions usually adopts a model belonging to one of rural, urban or freeway type. The solutions of the proposed techniques to evaluate the performance of the safety and infotainment type applications over 802.11p based VANETs would be neither complete nor credible, if due consideration is not given in the selection of appropriate traffic environmental models [122]. Accurate simulation of wireless vehicular communications for large-scale system testing is heavily dependent on the precision of the traffic environmental models. Incorrect assumptions made in these models can lead to erroneous conclusions on the effectiveness of any proposed techniques [123]. Researchers have broadly classified the road traffic environmental models [124] into three categories:

Urban Environment Model: The urban model is characterized by large number of MNs, dense signals, and greater number of junctions. Inconsistent mobility models with lower speeds owing to the existing traffic need consideration. The loss in communications between the RSUs due to the presence of buildings or other obstacles - an important factor for performance analysis in VANET - also needs to be taken into account. The research work presented in [125][126][127] highlights the effect of urban environment on the 802.11p communication systems.

Highway Environment Model: Higher vehicular speeds result in loss of services or intermittent signals [128][129]. The mobility models considered in these environments are fairly uniform. Efficient delivery of critical or prioritized messages (due to accidents, speeding etc.) is of high importance. The RSUs in such an environment do not experience great signal losses due to the absence of large or dense buildings or other obstacles. However sparse vehicles in this model have low throughput in V2V mode of communication.

Rural Environment Model: The rural environments are characterized by skeletal infrastructure [130] with narrow roads, fewer junctions and relatively uniform speed of vehicles. These aspects and the effects of communication loss due to terrain obstacles are taken into account.

CVIS [123] have used equipment to measure and study the effects of the 802.11p communication channel in urban, highway and rural traffic environments. Measurements are carried out based on the packet transacted data and RSSI. The results presented in that paper shows the effect of the environment on Packet Loss Rate and Network Throughput. The effects of vehicular speeds are not clearly described in that research work.

Bilgin and Gungor [124] have compared IEEE 802.11p with the IEEE 802.11b for V2V networks in different environmental conditions. The mobility models and the simulations tools that are used to simulate VANET are also discussed. The performance comparisons are carried out considering end-to-end delays, delivery ratios and network throughput. The results presented prove that the IEEE 802.11p is more efficient and outperforms IEEE 802.11b protocol for safety and non-safety applications operating under urban, rural and highway traffic environment scenarios. Thus, to evaluate the performance of transport and MAC layer protocols using important metrics like throughput, E2ED and packet drop, consideration of at least one traffic environment model is mandatory for VANET simulation.

3.2 Vehicular Mobility Generator

In order to implement traffic environment, a vehicular mobility generator simulation tool is used. Vehicular mobility generator is a road traffic simulation tool which generates real behaviour of vehicles with respect to different traffic conditions. Simulation of VANET scenarios is concerned with the precise modelling of radio signal transmission between vehicles, and between vehicles and RSUs under different traffic conditions, and for that one needs accurate position of vehicles under simulation.

Based on vehicular movement pattern and environment conditions, mobility generator produces trace files that capture realistic vehicular movement. The generated trace file is fed into the network simulator which specifies the realistic movement of each vehicle during the network simulation. The network simulator then implements the VANET protocols and generates its own trace files which consists of detailed information about each event taking place in the selected traffic environment. Gathered information is then analysed to evaluate the overall performance of the IEEE 802.11p in VANET. Some of the traffic mobility generator tools available are: MobiSim [131], VanetMobiSim [132], SUMO [133], MOVE [134], STRAW [135], FreeSim [136] and CityMob [137].

3.3 MobiSim Mobility Simulator for VANET

In order to get a convincing VANET simulation an appropriate vehicular movement pattern should be chosen [138][139]. The networking community has recently realized the importance of realistic mobility model for VANET. The majority of the literature on topics of mobile and vehicular networks utilizes simplistic and inaccurate mobility models. Only the last few years and in the latest works, the mobility models employed on vehicular networks use proven approaches of vehicular traffic flow theory that realistically reflects on the performance of networking protocols and techniques. Mobility model used by the mobility simulator provides realistic traffic behaviour that is input for VANET protocol simulation which determine if the proposed protocol is suitable for real implementation. To date, various mobility models have been developed for both MANET and VANET simulation, such as Random Walk model, Freeway model and Manhattan model. These models are incorporated and implemented in mobility simulator like MobiSim for traffic simulation.

MobiSim supports various mobility models for *ad hoc* networks including VANET, such as Random Waypoint Mobility Model, Random Walk Mobility Model, Probabilistic Random Walk Mobility Model, Random Direction Mobility Model, Freeway Mobility Model, Manhattan Mobility Model and Gauss-Markov Mobility Model. To successfully simulate realistic mobility scenarios, these mobility models require several input parameters, such as simulation duration, mobile node velocity, node number, direction, network area, obstacle and distance.

Mobility models are characterised by mobility pattern which can be categorised into two ways: macroscopic and microscopic [138][140]. Macroscopic mobility deals with aspects that influence vehicular traffic i.e. road topology, traffic intersection crossing rules, traffic signal, road lane, overtaking rules, traffic sign description establishment etc., whereas microscopic mobility deals with individual vehicle/driver behaviour interacting with other vehicles or with road infrastructure such as: travelling speed under different traffic conditions; acceleration, deceleration and overtaking criteria; behaviour in the presence of road intersections and traffic signs; general driving attitude related to drivers’ age, sex or mood etc.

A microscopic mobility pattern is mostly needed for analysing safety kind of applications, such as: lane changing warning, sudden breaking, blind spot warning and forward collision warning. Also, a macroscopic mobility pattern is recommended for non-safety types of applications because it directly affects protocol behaviour. Both these mobility patterns are needed for developing specific kind of applications.

For the present study, only a simple vehicular traffic flow in a freeway scenario using Freeway Mobility Model is considered and this is described in section 3.4.

Figure 3.1 shows MobiSim mobility simulator pluggable architecture which is composed of many modules and a form generator, in order to successfully simulate mobility in both MANET and VANET environments. Users can generate customized “mobility scenarios”, simulate them in graphical or batch modes, re-simulate the generated traces, evaluate measures on the traces, create diagrams from the evaluations, and finally test mobility model recognition algorithms on the traces [131].

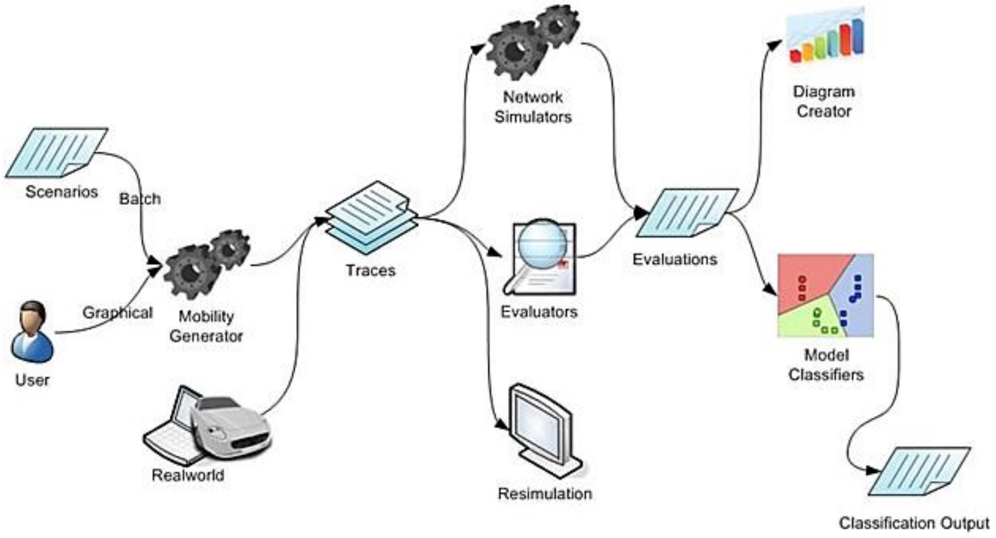


Figure 3.1: MobiSim pluggable architecture

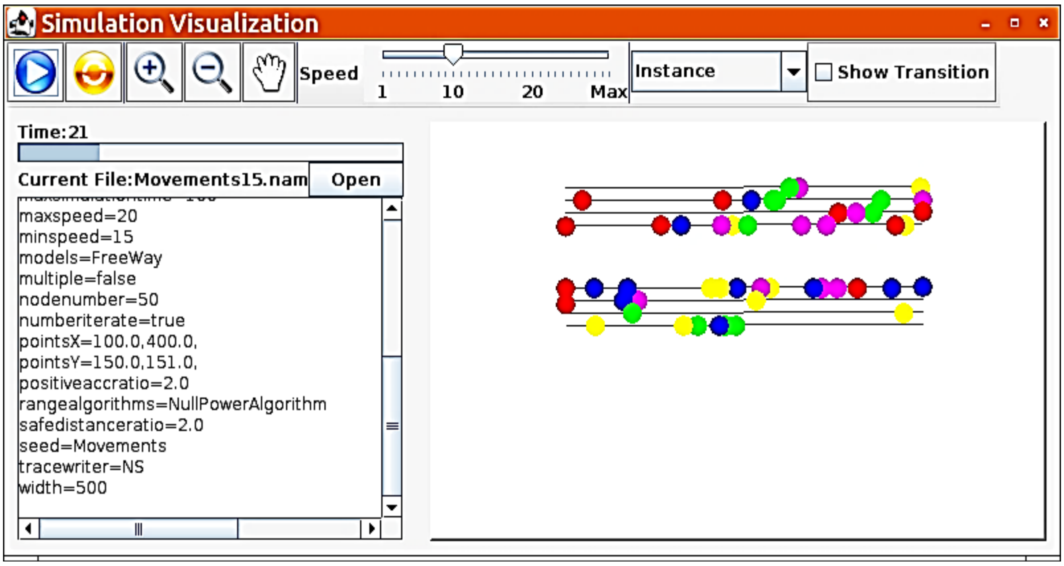
3.4 Freeway Mobility Model Using MobiSim

The Freeway Mobility model is designed to simulate mobility pattern of vehicles on a highway scenario [131][141]. It can be implemented in both microscopic and macroscopic node mobility patterns for various safety and non-safety applications. A freeway mobility model as defined in the map can have a number of lanes in both the directions without any intersection. The distinguishing characteristics of vehicles on a freeway can be summarised as below:

- Over a long distance, vehicles are restricted to its lane.
- The current speed of the vehicle is dependent on its previous speed.

- Strict geographic restriction on vehicular movement is imposed by not permitting a vehicle to change its lane.
- If two vehicles are on the same lane maintaining safe distance between them, then the speed of the trailing vehicle cannot exceed the speed of the leading vehicle. This relationship brings high temporal dependence to the freeway mobility model.

The proposed freeway VANET scenario considered in this study has two parallel highways with 4 lanes down traffic and 4 lanes up traffic. Figure 3.2 illustrates MobiSim interface showing the customized mobility scenario using freeway mobility model. This instance of freeway scenario consisting of a total of 50 vehicles in both directions will be used throughout in this work.



Time:21

Current File:Movements15.nam

```
maxspeed=20
minspeed=15
models=FreeWay
multiple=false
nodenumber=50
numberiterate=true
pointsX=100.0,400.0,
pointsY=150.0,151.0,
positiveaccratio=2.0
rangealgorithms=NullPowerAlgorithm
safedistanceratio=2.0
seed=Movements
tracewriter=NS
width=500
```

Figure 3.2: MobiSim showing a 8-lane freeway traffic scenario

3.5 IEEE 802.11p Based VANET Simulation using NS2

A network simulator is usually used to test and evaluate protocol behaviour because arranging a real test bed is expensive and time consuming. Using network simulator, both wired and wireless network can be tested under different network conditions. In general, it performs packet level simulation of source, destination, data traffic transmission, signal reception, link between nodes and channel access. In contrast to real test bed, which incurs cost and time, network simulators almost do the same job with ease and within budget [138][140] .

There are various flavours of network simulators such as NS2, NS3, GloMoSim, MATLAB, QalNet, CAVENET, OMNeT++ and OPNET. Among all these, NS2 is most trusted and widely accepted by research community in the last two decades. Kurkowski et al [142] survey showed that NS2 is the most preferred simulator for *ad hoc* network research.

In this research work, network simulator version NS2.35 is used. The NS2 simulator was originally intended for guided media but the Monarch Group at CMU have enhanced NS2 to support wireless networking such as wireless LANs, MANET as well as VANET [143][144]. It implements various networking protocols (TCP, UDP), data traffic sources (FTP, PBC, CBR), queue management mechanisms (Queue/DropTail/PriQueue), routing protocols (AODV, DSDV, Dijkstra) etc. NS2 is written in C++ programming language and OTcl scripting language to separate the control and data path implementations. The simulator supports a class hierarchy in C++ (the compiled hierarchy) and a corresponding hierarchy within the OTcl interpreter (interpreted hierarchy).

NS2 has features to support wireless network standard, specifically MANET, but it needs VANET extension. Though it has built-in support for 802.11p protocol (from NS-2.33 onwards), it has no vehicular traffic flow model and no radio signal obstacles. Due to its portability and ease of setup, a mobility simulator like Java based MobiSim can be easily extended to couple the network with VANET mobility generator. In addition, this simulator is a freeware, open-source and has many different examples available and a huge on-line forum support [138][139]. The block diagram Figure 3.3 shows how network simulator NS2 interacts and interfaces with mobility simulator MobiSim.

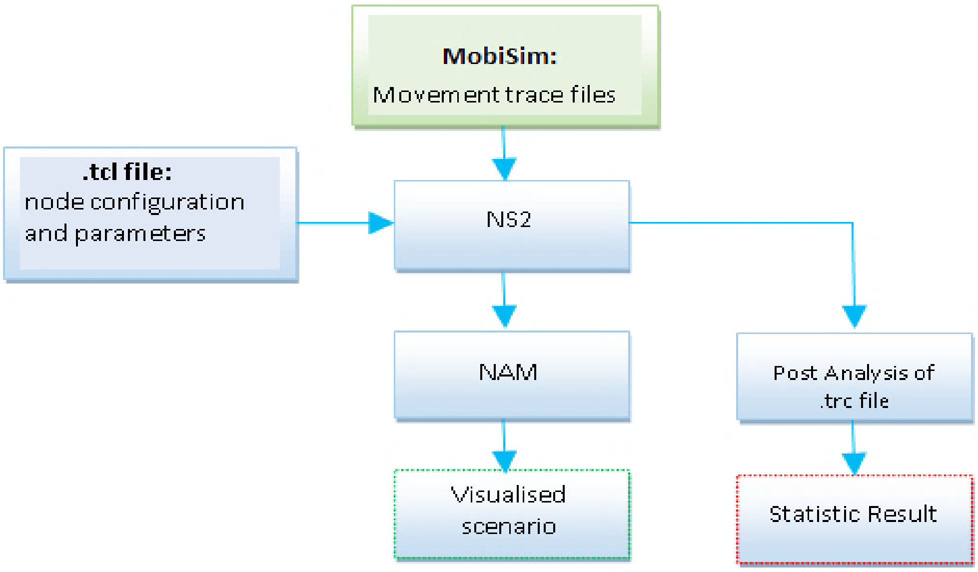


Figure 3.3: NS2 and MobiSim interaction process

3.5.1 DSRC's Modulation Scheme in NS2

DSRC implementation in NS2 is achieved by applying the IEEE 802.11p MAC and PHY layers features in the TCL simulation code. The NS2.35 version has introduced two new modules: *Mac802.11Ext* and *Wireless-Phy-Ext*. These two extensions are based on the default Mac802.11 and WirelessPhy, which are able to simulate DSRC's MAC frame in different modulation schemes.

Modulation schemes are implemented at PHY layer of wireless network to control data flow rate. Modulation influences throughput of the communicating nodes and this reflects on QoS. Four types of modulation schemes are available with IEEE 802.11p: BSPK, QPSK, 16-QAM and 64-QAM. The following Table 3.1 shows them, keyed with the reference index used in NS2.35. DSRC's MAC frame modulated with high modulation index has shorter transmission duration but requires higher SNR to receive and decode it. BPSK and QPSK provide low data rates but can cover a wider range with low SNR. NS2.35 users can define singularly or adaptively all these modulation schemes for the entire simulation.

Modulation Index	Modulation Scheme	Data Rate in Mb/s
0	BPSK	3
1	QPSK	6
2	16-QAM	12
3	64-QAM	27

Table 3.1: Modulation index used in NS2.35 for 802.11p simulation

3.5.2 Data Traffic Source for Safety Application

A VANET requires exchange of periodic broadcasting of safety messages among the vehicles for various cooperative awareness applications. These periodic messages are well known as BSM in the U.S.A. and CAM in Europe [97]. In network communications, a traffic load is generated to carry information from source to destination. This research work requires both BSM and TCP based non-safety data traffic source for simulation. In vehicular wireless communication, the main traffic load is beacon based and it broadcasts data packets to provide safety services to one-hop neighbouring vehicles. Periodic broadcasting of safety messages in the VANET is essential to serve the core purpose of DSRC, even in the presence of other data traffic such as TCP based file transfer application.

For DSRC safety messaging simulation in NS2, a specific PBC agent was developed by Chen et al [145], which was jointly designed by Mercedes Benz Research & Development North America, Inc. and University of Karlsruhe (TH). This PBC agent can be found in IEEE802.11p protocol extension in NS2, which allows users to develop simple short messaging applications for VANET. The agent PBC defines message generation frequency, the frame data modulation scheme and MAC frame payload size, so that one can approximate a real safety messaging applications in VANET environment.

The TCL name of the agent is Agent/PBC. Packet type is PT_PBC. Some examples of the TCL commands [145] to define the message generation behaviour are as follows:

```
Agent/PBC set payloadSize 100; # (in bytes)
Agent/PBC set periodicBroadcastInterval 0.05; # (in seconds)
Agent/PBC set periodicBroadcastVariance 0.05; # (in seconds)
Agent/PBC set modulationScheme 0; # the default modulation scheme for data is BPSK
```

The parameters of the Simulated Safety Messaging Application in NS2 code are:

PeriodicBroadcastInterval: 0.05, 0.10, 0.15, 0.20
ModulationScheme: 0, 1, 2, 3

3.5.3 DSRCs Multi-channel Support in NS2

NS2 can simulate vehicular wireless network but it is not fully adapted to DSRC based VANET, specifically in respect of IEEE 1609.4 multi-channel operation. Some of the researchers have attempted to extend NS2 to support multi-channel operation capability, but till now none of them have successfully achieved the standard as defined in IEEE P1609.4.

Q. Chen et al [64], have analysed implications of the IEEE 1609.4 multi-channel operation on vehicular safety related applications by adding certain features in NS2. They have considered only CCHI transmissions in their scenario but without including non-safety application over SCHI transmissions. K. Hong et al [146] tried with little improvement to evaluate the performance of multi-channel operation for only safety applications using NS2 simulator. However, their multi-channel operation scheme does not meet the required IEEE 1609.4 standard. They managed to use basic safety messages only.

Y. Chen et al [147] proposed an obstacle model integrated with NS2 to support VANET simulation. However, their model does not consider IEEE P1609.4 multi-channel operation standard. Ghandour et al [148] implemented IEEE 1609.4 multi-channel operation simulation tool in NS2. They have tried both CCHI and SCHI operations; however, their tool does not conform to the realistic multi-channel simulation as defined by DSRC standard. Their tool performs by some queues at the Interface Queue (IFQ) layer of NS2 mobile nodes, and it does not exhibit a genuine modelling of wireless multi-channel operation based on DSRC-IEEE1609.4.

Torabi et al [149], implemented new WAVE tools in NS2 that realistically model DSRC multi-channel operation at the PHY layer (i.e. at the Network Interface (NetIF) and Channel layers of NS2 mobile node architecture) in addition to the IFQ layer. They compared their tool with the work of Ghandour et al and showed performance improvement with experimental analysis. Despite successfully achieving better PDR, their tool is not integrated with the IEEE 802.11p EDCA mechanism, and also does not give support to upper layers of DSRC.

As far as we know, there is not a single publicly available simulation tool that genuinely simulates complete DSRC's protocol suite as draft-standardized by IEEE.

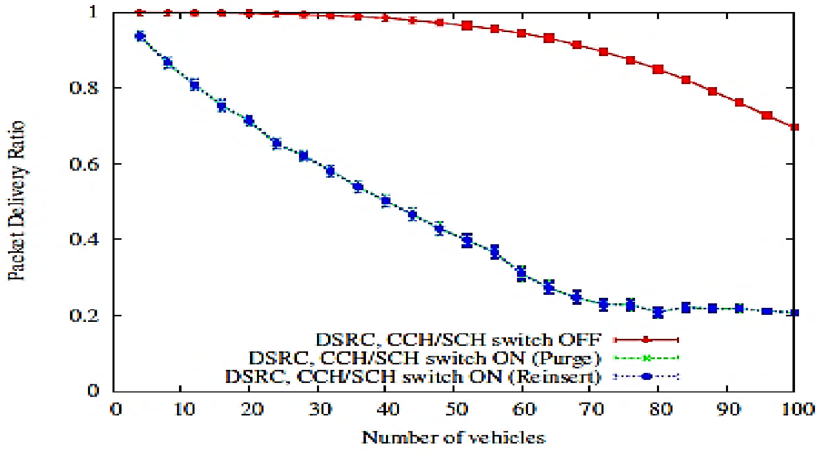


Figure 3.4: The packet delivery rate [148]

CCH/SCH Switch OFF: A scenario in which all the vehicles are always tuned on the CCH channel and no channel switch occurs is interesting. This configuration reflects the current status of the stack in NS2, used by most researchers to simulate VANETs environment. Ghandour et al, have used this configuration as a benchmark to analyse the impact of the strict synchronization and periodic channel switching imposed by the IEEE 1609.4 protocol.

In this research work NS2.35 simulator is used, which by default supports single channel operation, where both BSM and TCP based non-safety applications co-exist and there is no channel switching; this assumption can be seen in Figure 3.4 as **switch off** curve. This figure shows a trend similar to that in Figure 4.4 (PDF at modulation 0) in Chapter 4, which confirms the accuracy of the present simulation result and setup. Similar simulation approach and setup have been carried out by several other VANET researchers including Acatauassu et al [58] and Ghandour et al [148].

3.5.4 NAM Output in NS2

The Tcl/Tk based animation tool Network Animator (NAM) is used to visualise NS2 simulation while it is running. The generated NAM trace file contains various information, such as network topology, node links, packet, queues, number of nodes, node connectivity and packet trace information. NAM event has a fairly simple and consistent structure and it can have various visualization modes. Time line feature in NAM allows the user to move forward and backward and locate a specific simulation point as well as control the simulation running speed.

Figure 3.5 shows NAM output having the same 50 vehicles, as generated using MobiSim shown in Figure 3.2. Here the small blue squares represent the vehicles and the red hexagon in the middle represents one RSU. The black circles in the NAM shows message transmissions from different vehicles as well as from RSU.

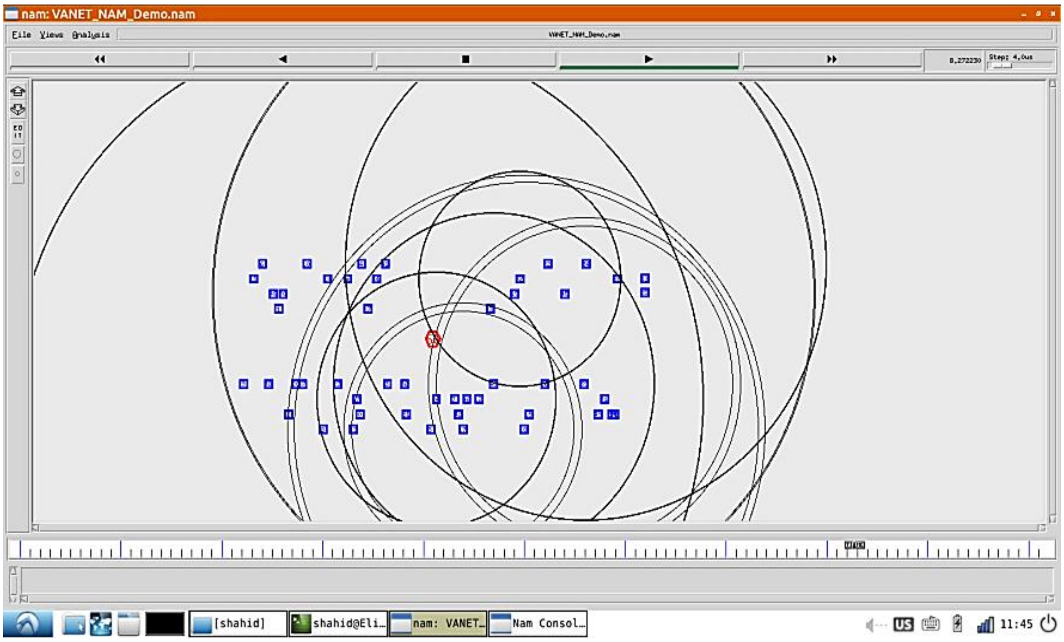


Figure 3.5: NAM showing 50 vehicles on a bidirectional freeway scenario

3.6 Routing Protocol in Freeway Scenario

VANET requires fast, reliable and seamless V2V or V2I communication. When the signal gets out of range, the vehicle's data traffic is routed in multi-hop fashion in order to access resources from RSU or other vehicles. In a multi-hop network, intermediate nodes act as a relay to route data from source to one or many destinations. Routing protocol plays a very important role in protocol performance, if the data are to be delivered in multi-hop fashion.

It is demonstrated in the research work [150] that choosing routing algorithms has direct bearing on TCP behaviour in MANET. VANET requires robust routing protocol that has the capability to adapt to dynamic network condition, has connection stability and reduced discovery overhead, and is able to find path before the connection breaks. There are plenty of routing protocols available, but a routing protocol for VANET environment should possess the following general distinguishing characteristics [151] in order to provide efficient support for both beacon and TCP based applications:

- Adaptive to highly dynamic topologies
- Connection stability
- Low latency for neighbour discovery
- Low control overhead
- Loop-less routing path
- Low complexity
- Short convergence time
- Scalability and Multicast capabilities

The topology-based routing protocols, such as reactive AODV, DSR and DSDV that are often used in the MANETs, can also be used in the VANETs [114][152][153]. However, the main challenge in the deployment of those protocols is that the overhead incurred in path discovery and path maintenance can be rather high due to high mobility. Such protocols are mainly deployed on highways scenarios or a small-scale network with small number of hops between source and destinations.

Due to high topology variance in a freeway environment, vehicles in V2V or V2I multi-hop communication mode are not always coordinated to each other. For freeway scenario AODV is used, because various studies [153][154][155][156] have suggested that AODV in comparison to DSDV, DSR and OLSR is best suited for VANETs freeway environment, where vehicular density is sparse and vehicular speeds are high. The other reason for using AODV is that when a vehicle moves out of range of the vehicle with which it is currently in association, it receives notice of the failed transmission from the link layer. This triggers an immediate search to find a new route, which as a rule, takes less than 50ms to complete. But DSDV does not use Wi-Fi link-layer feedback, and it does not search for a new route until the very next regular scheduled round of distance vector commences, which might be delayed many seconds.

AODV [157] is a reactive on-demand routing protocol with small delay, which finds route of destination only when it has data to send in order to reduce traffic overhead. AODV uses Bellman-Ford distance vector routing algorithm adapted to mobile scenario, and it is considered as the most suitable routing protocol for MANET. It is motivated by limited bandwidth that requires to maintain the route once established, as long as unicast or multicast routing is needed. AODV has also resolved Count-To-Infinity and loop problems by introducing sequence number and registering cost. It greatly differs from other reactive protocols in using Destination Sequence Number (DesSeqNum) to find the latest route to destination. AODV routing goes through three distinct phases: route discovery, data transmission and route maintenance.

AODV uses two control packets for route discovery: the Route Request (RREQ) packet is broadcast by the source vehicle which is stored in the intermediate vehicle's OBU, and the Route Reply (RREP) packet in response to RREQ, going in the reverse way, is unicast to source vehicle. There is also a Route Error (RERR) control packet which is used to inform link failure during route maintenance.

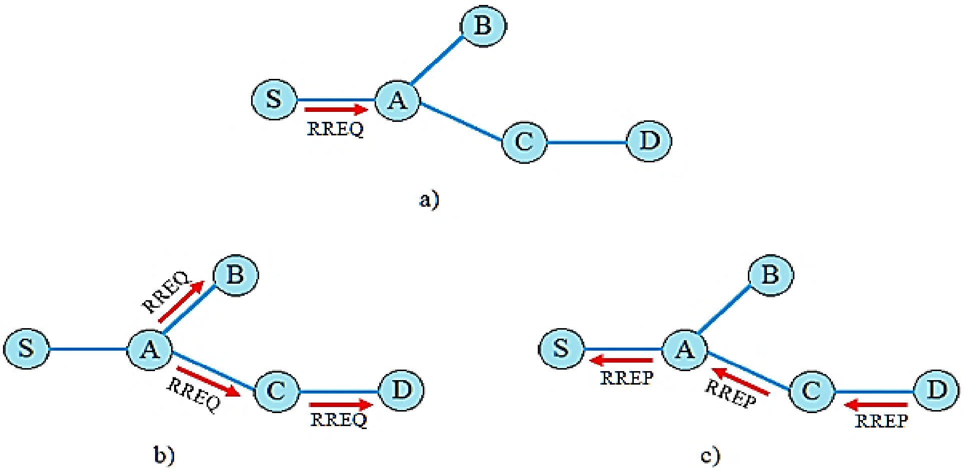


Figure 3.6: AODV Route discovery mechanism

Figure 3.6 shows AODV route discovery mechanism: a) Source node 'S' has data to send to destination node 'D'; so it initiates path discovery process by sending RREQ packet to its immediate neighbours. b) Node 'A' broadcasts RREQ packet to all its neighbours' nodes. c) Destination node 'D' send unicast replies back to source node 'S' via two intermediate nodes 'C' and 'A'.

3.7 Propagation Model for Freeway Scenario

In a real vehicular wireless communication, a vehicle may experience packet drops due to signal attenuation caused by multi-path signal propagation, reflection & refraction from obstacles, signal distortion and other adverse environmental conditions. Therefore, simulating vehicular wireless communication requires detailed representation of realistic radio propagation in any given specific environment, freeway included.

VANET, when operating in infrastructure-less mode, depends on distributed control to regulate wireless channel access. The protocol performance, when deployed in real world, may vary from simulation results, because simulator might behave in an excessively affirmative or conservative manner. For example, in a real scenario, it may happen that two communicating nodes manage to exchange information in a poor reception condition, which may not happen in the simulated situation because of some inaccuracies in the propagation model.

The selection of a radio propagation model has huge impact on the performance of a routing protocol, because the propagation model determines the number of vehicles in the collision domain, which is a crucial input for interface as well as channel contention. Also, it has direct impact on vehicle’s ability to send a packet to neighbouring nodes. In a VANET, a vehicle may frequently and intermittently join and leave each other’s transmission range. Depending on the propagation model’s predicted received signal power, a vehicle may share a collision range with a few or as many as 50 of the other vehicles. This happens because the model takes into consideration the presence of road side obstacles such as trees and buildings, which reduce the Line of Sight (LoS) of the vehicles.

Network simulator NS2 implements radio propagation model which account for packet loss in the wireless medium. When a packet is received, the signal power associated with each packet is compared with the receiving threshold. The received packet is considered to be correct, if the signal power exceeds the receiving threshold; otherwise at MAC layer it is discarded. NS2 supports various propagation models including Nakagami.

The Nakagami fading model is used in the simulation setup, because it is suggested by many authors [158]–[161] as a suitable channel model for freeway scenario based on empirical data. Taliwal et al [158] empirically tested and recommended Nakagami fading model for the vehicular freeway scenario. This is a mathematical modelling of radio channel and fading, suitable for outdoor medium. In comparison to log-normal shadowing and two-ray ground models, Nakagami propagation model has more options to configure parameters to facilitate better representation of moderate wireless channel fading in a freeway environment.

Nakagami distribution is defined by the following probability density function [145]:

$$f(x) = \frac{2m^m x^{2m-1}}{\Gamma(m)\Omega^m} \exp\left[-\frac{mx^2}{\Omega}\right],\ x \geq 0,\ \Omega > 0,\ \mathrm{m} \geq 1/2 \qquad (3.1)$$

The corresponding probability density function of power at a given distance can be obtained by a change of variables and is given by a gamma distribution as mentioned in equation (3.2).

$$p(x) = \left(\frac{m}{\Omega}\right)^m \frac{x^{m-1}}{\boldsymbol{\Gamma}(m)} \exp\left[-\frac{mx}{\Omega}\right], x \geq 0 \qquad (3.2)$$

where Ω in the equation (3.2) is the expected value of distribution, which defines the average received power at a given distance whereas *m* defines fading intensity which depends on distance to the sender. Hence, the Nakagami model is defined by two functions: Ω (d) and *m*(d). So, it can be concluded that the Rayleigh distribution is a special case of Nakagami distribution where *m*(d) =1 for every d, and the larger value of *m* indicates less severe fading.

The implementation of Nakagami RF model in NS2 is specified in *nakagami.cc* file and its TCL class name is *Propagation/Nakagami*, which defines a set of parameters to control RF signal propagation that includes fading and attenuation.

3.8 Performance Evaluation Metrics

In this work, when comparing the protocol performance of both BSM and TCP based file transfer applications, the following performance metrics are used to study and evaluate them:

Throughput: Average rate of successful transmission of packets to destination STA in the network defines the term throughput. Effective utilization of bandwidth is measured through throughput/consumed bandwidth, where the higher the throughput, the better the Quality of Experience (QoE). It is one of the key metrics used to measure how well an application is performing. Required throughput threshold varies from application to application. The throughput is equally essential for both safety and non-safety applications that run over VANET. Application usefulness and technology effectiveness may be called into question if the required minimum threshold level of throughput is not achieved. A variety of factors affect the throughput, such as BER, RSS, density of vehicle on a single AP, vehicular velocity, link failure because of high speed, packet drop, congestion, multi-path signal fading and frequent channel hand-off. Even though DSRC achieves promising low latency performance, the throughput needs further improvement.

Together with packet delay, the throughput is the most important aspect in the assessment of the vehicular wireless network performance. Generally speaking, it is challenging to get both high throughput and small packet delay. In most cases, achieving a maximized throughput would have a considerable impact on the average number of end-to-end delay that causes overall QoE degradation. Without acceptable throughput and minimum delay, both safety and infotainment applications may not work as expected.

Packet Drop: There could be several reasons for an on-the-fly packet going nowhere, such as network congestion, packet corruption due to signal interference (a very common case in wireless networks), request time-out because of multi-path propagation delay, channel handover or broken next hop link. The packet drop is measured by counting the total number of transmitted packets dropped during transmission from source vehicle to destination vehicle. The lower the packet drop, the better the performance of the system in terms of both throughput and PDF.

Average End-to-End Delay (E2ED): Average latency a data packet experiences to successfully reach from source to destination vehicle is measured in terms of E2ED. This latency includes all possible delays caused by buffering during route discovery, queuing at the IFQ and re-transmission delays at the MAC layer, propagation delay as well as transmission times. Safety applications have strict constraints on maximum tolerable delay. For instance, time-critical safety applications like vehicular collision avoidance, intersection collision warning, and lane changing warning require E2ED to be less than 100ms. But non-safety TCP/UDP based applications such as interactive gaming can tolerate up to 150ms of E2ED, while bursty applications such as video or audio streaming can tolerate up to several seconds of delay but higher delay reduces the throughput, and of course the performance of the system as well [162].

Packet Delivery Fraction (PDF): It is the ratio of the number of data packets delivered to destination vehicle to those created by the source vehicle. The higher the PDF, the better the routing protocol. When the throughput of the system is too marginal to make a judgement call, then it is best to evaluate protocol performance based on PDF.

3.9 Simulation Scenarios and Experimental Parameters

In this research work, two simulation scenarios are constructed. The Scenario-1 is used to study performance evaluation of safety messaging application in V2X communication mode, considering two important parameters, namely modulation schemes and broadcast message interval rate. The Scenario-2 is taken up for studying TCP based non-safety application coexisting with BSM over DSRC.

3.9.1 Simulation Scenario-1

This Scenario-1 starts with 10 vehicles and in increments of 10 it reaches the maximum of 50 vehicles in a bidirectional 8-lanes ideal freeway within the vicinity of single RSU. Generally, freeway is a sparse vehicular network, where the density and frequency of vehicle is much less than urban busy environment. The vehicular density impacts the performance of the protocol but one can better estimate protocol performance based on medium vehicular velocity of 54 – 72 km/h, medium density of 10 – 50 vehicles as well as recommended BSM data payload of 100 bytes. There is a single RSU that has the same simulation parameters as defined for vehicles, including omni-directional Non-Line of Sight (NLoS) of 250-meter signal range. In this scenario, both vehicles and a RSU are periodically broadcasting BSM at varying intervals to one-hop neighbours. For vehicular safety applications, the performance of all four modulation schemes and varying message interval rates are studied with respect to increasing number of vehicles. The relevant parameters and their values are displayed in Table 3.2.

Parameters	Value
No. of Vehicles	10 20 30 40 50
Max Speed of Vehicles	72 km/h
Min Speed of Vehicles	54 km/h
Safety Application Agent	PBC
PBC Payload Size	100 bytes
PBC Interval	0.05, 0.10, 0.15, 0.20 and 0.25s
PBC Variance	0.05s
Communication Protocol	802.11p
Frequency	5.9 GHz
Channel Width	10 MHz
Communication Range	250 metres
Antenna Type	Omni-Directional
Propagation model	Nakagami
Modulation Index	0, 1, 2, 3 (BPSK, QPSK,16QAM,64QAM)
MAC	Mac/802_11 Ext (Not used p1609.4 & EDCA)
PHY	Phy/WirelessPhyExt

Table 3.2: Scenario-1 simulation parameters for RSU & vehicles

3.9.2 Simulation Scenario-2

Scenario-2 is similar to Scenario-1 except for a few changes in some of the simulation parameters. This is used in studying the performance of both safety and non-safety applications. Since safety messaging application is integral to data flow in DSRC based VANET, in this scenario, bandwidth is co-occupied by both safety and non-safety data traffic. Some of the vehicles in the scenario are accessing FTP resources from RSU in a multi-hop fashion, whereas all the 50 vehicles are periodically broadcasting at the same time safety data to one-hop neighbours.

Performance analysis result of safety messaging application as studied in Chapter 4.1 using Scenario-1 proved that the BPSK modulation scheme is most optimal and reliable with respect to PDF, throughput and E2ED metrics, and of course, for the freeway VANET on the whole. Hence, based on the simulation study of Scenario-1, BPSK modulation scheme that has 3 Mbps data rate is chosen for Scenario-2. The message interval rate for BSM is kept varying in order to analyse the optimum performance of both safety and non-safety applications, when they co-exist on the channel. Table 3.3 provides the associated data.

Parameters	Value
No. of Vehicles	50
Max Speed of Vehicles	72 km/h
Min Speed of Vehicles	54 km/h
Safety Application Agent	PBC
PBC Payload Size	100 bytes
PBC Interval	0.05, 0.10, 0.15, 0.20 and 0.25s
PBC Variance	0.05s
Transport Agent	TCP
No. of TCP Flow	3
TCP Variant	Reno, BIC, Scalable, Westwood, HighSpeed and LP
Application Agent	FTP
BDP	30000 bytes
MSS	4096 bytes
Routing	AODV
Communication Protocol	802.11p
Frequency	5.9 GHz
Channel Width	10 MHz
Communication Range	250 metres
Antenna Type	Omni-Directional
Propagation Model	Nakagami
Modulation Index	0 (BPSK)
MAC	Mac/802_11 Ext (Not used p1609.4 & EDCA)
PHY	Phy/WirelessPhyExt

Table 3.3: Scenario-2 simulation parameters for RSU & vehicles

3.10 Chapter Summary

This chapter has described traffic and mobility modelling using the tool MobiSim, followed by how VANET simulation is to be organised and carried out using NS2, the premier network simulation tool. Along the way, necessary details of different modulation schemes and some routing protocols supported by NS2, and the propagation model suitable for a freeway scenario were also given. Subsequently, the definition and significance of performance metrics, against which the simulation results are to be evaluated, were presented, The chapter closes with the description of two simulation scenarios along with their prescribed parameters, which are constructed in a way to effectively act as a kind of test bed or sounding board to experiment with the innovative ideas mentioned in this thesis. Taken together, this chapter has provided the necessary background and configured the settings for the simulation experiments to be taken up later. In the following Chapters 4, 5 and 6, which form the core of the present dissertation, the findings of this simulation study on the performance of both BSM application and TCP-based file transfer application will be comprehensively discussed and critically evaluated.

Chapter 4

Performance Analysis of BSM and Non-safety Applications

This chapter begins with section 4.1 where simulation studies are conducted to examine the performance of BSM Application over DSRC based VANET, and to find the optimum broadcast message interval rate and a suitable modulation scheme. Taking into consideration the optimum message interval over suitable modulation, simulation studies are carried out, in section 4.2, to assess the performance of TCP based FTP application. It is necessary to present and discuss these preliminary results from this simulation so that the real and *hidden*, if any, characteristics of VANET are clearly understood and only then can one appreciate how hard it is to use TCP under highly congested VANET scenario. Finally, in section 4.3, the impact of TCP over BSM is studied under optimum modulation scheme.

4.1 A Study on the Performance of DSRC Safety Messaging Application

It is difficult to establish ITS without the use of wireless network, especially VANET. VANET communication is an important way to help drivers get time sensitive safety information, such as collision warning, abrupt lane changing warning etc. IEEE adopted Wi-Fi based 802.11p communication standard for vehicular communications that guarantees delivery of fail-safe safety message on time, and whose performance is dependent on various parameters, including message payload and the transmission interval rate. The message payload associated with the modulation and coding increases the data rate and the average throughput of the wireless network, while the transmission interval is related to the re-transmission rate and the probability of correct and timely reception of the transmitted message. These parameters affect the reliability and efficiency of 802.11p based V2X communication system.

In this phase of the work, the performance of BSM application over 802.11p based VANET system with message payload and transmission rate are investigated to find optimum broadcast message interval rate over a suitable modulation scheme. The performance is evaluated with respect to the metrics, PDF, E2ED and throughput, by varying modulation schemes and transmission interval rates. The experimental results show that the correlation of modulation and the transmission interval allows for selection of the appropriate combination for optimum results, based on BSM application criteria.

4.1.1 Introduction

In VANET, there are different DSRC applications designed for different purposes. The message payload and the frequency are the more important parameters that will depend on the purpose of the application, which in turn decide their performance. In this work an extensive simulation study is made on the performance of a typical BSM application by varying some of these important parameters and the results are presented here. The preliminary results obtained bring out the true behaviour and real characteristics of VANET.

The two parameters that critically affect the performance of safety applications and have not been widely studied in the literature are the message payload and the transmission interval rate. The message payload, through the modulation and coding that is imposed on the system, dictate the data rate and consequently, the maximum throughput of the wireless network. The volume of information that can be embedded into safety messages is directly related to the modulation used for their transmission: the higher the modulation index, the more the safety information can be transmitted and the safer the system becomes. In the same way, the transmission interval is one-to-one related to the re-transmission rate and the probability of correct reception of the transmitted message.

Rapid VANET topology impose strict timing requirements. The timing requirements can be appreciated from the fact that it is only necessary to communicate about any upcoming dangerous situation, before the situation turns into a reality and perhaps can be avoided (e.g., communicate a probable collision before the vehicles collide). Therefore traffic safety systems could be classified as real-time systems [61].

Real-time communication implies that there should be an upper bound on the communication delay which must be smaller than the deadline. In a hard real-time system, if the correct data does not reach its intended recipient before a certain deadline, the data is futile and the missed deadline will lead to, to varying degree, severe consequences for the system and application performance. When the transmission interval is low, the E2ED of the message - a metric that is so crucial for safety applications - is low too. The lower this delay, the more time will be left to drivers to react which will prevent likely accidents. Furthermore, a low E2ED satisfies the real-time communication requirements. Thus, it is clear that the effectiveness and reliability of a DSRC based V2V communication system is intimately connected with these two parameters.

In this study, the effect of variable message length in terms of modulation scheme and transmission interval rate is examined. Simulations are conducted using the NS2 simulator along with its extensions for DSRC systems for a freeway scenario. The performance metrics utilized are PDF, E2ED and throughput, and the results are obtained by varying modulation schemes and transmission interval. The correlation of the modulation and the transmission interval with E2ED and throughput is studied systematically through these simulations.

4.1.2 One Hop Broadcast for BSM

Beacon based BSM broadcasting plays many important roles in VANETs, especially for delay-sensitive applications which are critically important for the safety of vehicles. The safety short messages, like V2V collision avoidance, require periodic broadcasting at the average interval rate of 0.1s to avert any likely accident. The main function of periodic BSM broadcasting is to regularly estimate the current position of the vehicle that includes velocity, distance, direction, geo-location etc. and to broadcast time sensitive safety messages to one hop neighbours in order to prevent V2V collision. If the vehicles are periodically receiving safety messages from their neighbours, then they can decide the best possible course of action in time to avert possible accidents. Reliable and timely delivery of safety message on highly dynamic vehicular network presents several key challenges, such as harsh communication environment, frequent disconnection between peers and multi-path propagation delay.

In one hop broadcasting, the OBUs of vehicles do not flood the safety messages; rather, when a vehicle receives a safety messaging packet, it keeps the information in its on-board database. The vehicles then select some of the meaningful information from the database, analyse them, and then may automatically actuate or adjust some of the vehicles sensors to prevent any untoward incident or broadcast the same information to neighbours for various other purposes. Thus, with one hop broadcast, each vehicle will generate, store and carry real-time safety information with itself as it moves forward, and this information will be transferred to other vehicles in its one-hop neighbourhood in the next broadcast cycle [44].

4.1.3 Simulation Scenario

Scenario-1, as described in section 3.9.1, is considered to study performance of safety messaging application in VANET. The nodes containing safety messages are broadcasting at set times, at the rate of 0.05 – 0.2s and there are no other data traffic. The messages are broadcast messages and therefore no ACK in response is awaited. All vehicular nodes are using the IEEE 802.11p MAC as presented in chapter 2, and each node (vehicle) must listen during an AIFS before sending and back-off if the channel is or becomes busy during the AIFS.

As all the data traffic is comprised of broadcast safety messages, they carry the same highest priority status. In a real implementation where data traffic from different applications share the same communication channel, non-safety data traffic would have lower priority than safety messages. The case with a missing ACK will not appear since all data traffic is broadcast and there is no specific intended recipient. With this in mind, the transmitter will never make more than two attempts to access the channel in order to try to send a packet.

4.1.4 Simulation Results and Discussion

In this section, simulation results are presented on the performance of BSM application over variable modulation schemes and with different message intervals rate. The results are derived only with the safety application running in VANET (without any crossing TCP traffic).

4.1.4.1 Performance with respect to Modulation Schemes and with Number of Nodes

Figure 4.1 to Figure 4.5 present the simulation results obtained using different modulation and increasing number of nodes with respect to total sent packets, total received packets, PDF, E2ED and throughput ratio respectively, with the broadcast message interval being 0.05s for all four modulation schemes.

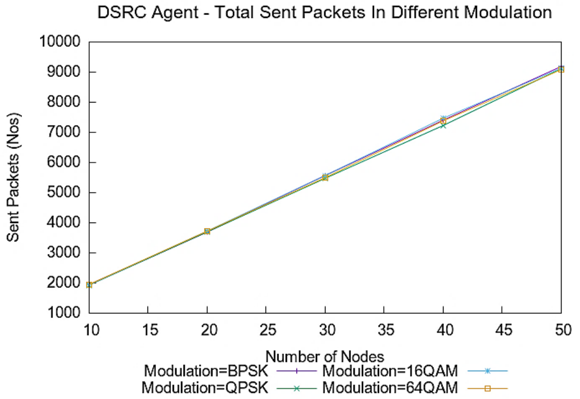


Figure 4.1: Sent packets in different modulation and No. of vehicles

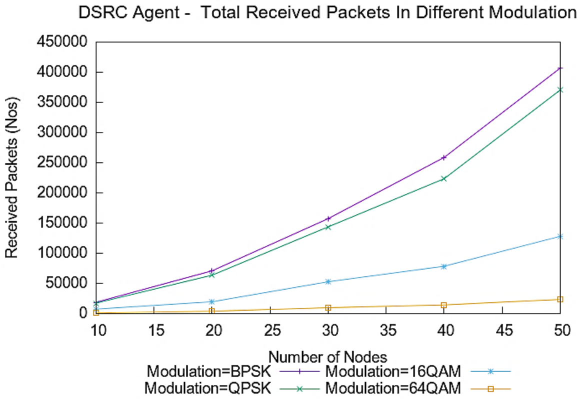


Figure 4.2: Received packets in different modulation and No. of vehicles

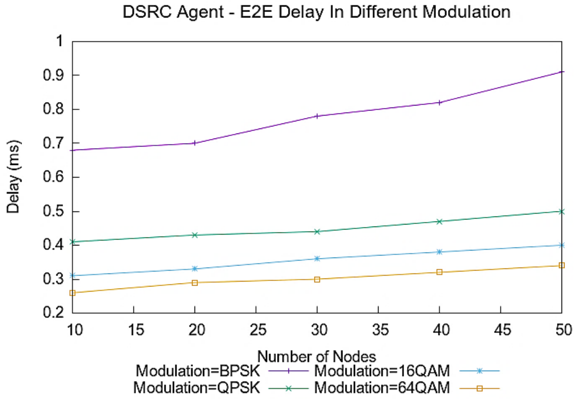


Figure 4.3: E2ED in different modulation and No. of vehicles

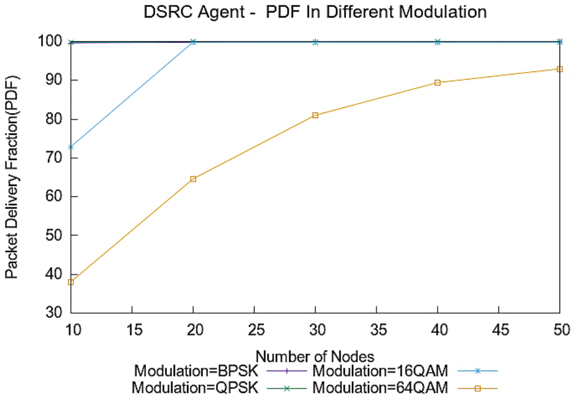


Figure 4.4: PDF in different modulation and No. of vehicles

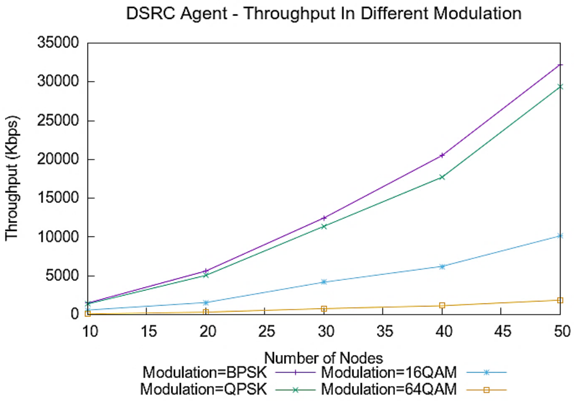


Figure 4.5: Average Throughput in different modulation and No. of vehicles

E2ED is the most crucial metric for the efficiency and reliability of safety applications and no packet should be delayed beyond a specified time limit to reach its destination. In case real-time constraints that are necessary for timely reflex of the driver are not met, the data will be useless and the excessive delay will jeopardize system performance.

The majority of the vehicular safety critical applications like V2V collision avoidance, lane changing warning, curve speed warning, pre-crash warning and intersection collision warning etc. require an average of 0.1s of message delivery delay, while for the non-safety applications an average of 0.15s is acceptable. From Figure 4.3, it can be seen that as the modulation type goes through from BPSK to 64-QAM, in the sequence as given in Table 3.1, the E2ED decreases, while the increase in the number of nodes of the system produces a relatively stable E2ED in each modulation. In every case, the delay is low enough for the driver to take timely action. The fact that the delay remains stable and steady with increasing number of nodes, is a very appealing feature, as this means that the performance is insensitive to the network's density, and in any case there will be no real-time violations.

Furthermore, the average E2ED is much lower than the previously mentioned fail-safe safety messaging application limits, in every modulation *vs* number of nodes case. More specifically, it ranges from 0.2ms for the 64-QAM case, to (0.7ms, 0.9ms) for the BPSK case. Thus, as 64-QAM scheme outperforms the other modulation schemes, it can be said that it is the preferred modulation scheme for applications that require the lowest E2ED. The suitability of the 64-QAM is further enhanced by the fact that when such a modulation is used, more valuable information can be transmitted, which in the case of safety messages, means that more safety measures can be undertaken.

In Figure 4.4 the PDF performance results are presented with different modulation schemes against number of nodes. It can be seen from this graph that BPSK and QPSK modulation schemes produce 100 percent PDF irrespective of increment in number of vehicles, which confirms their suitability as a reliable modulation scheme for safety applications. Theoretically speaking, higher modulation like 64-QAM indicates that more data can be transferred and therefore should be favoured for safety applications, if the distances between the vehicles are close. But, in low vehicular density scenario, PDF of 64-QAM scheme performed very poorly and this is mainly due to message lengths.

Throughput results are presented in Figure 4.5, where it can be seen that as the modulation type progresses in order (as given in Table 3.1), the throughput decreases, and when the number of nodes increases, the throughput increases too. As can be seen in Figure 4.2, the received packets of each node are considerably higher than the packets sent in Figure 4.1, because in the simulated scenarios every vehicle is broadcasting message in one hop to the network, so a vehicle is receiving more messages from all nearby vehicles than it is sending. As a result of this, the throughput increases along with the number of nodes.

Furthermore, from Figure 4.5, it can be seen that as the number of nodes in the system increases, the throughput of the system in all modulation cases gets higher. This is more obvious in the cases of lower modulation schemes, such as BPSK and QPSK. Despite having highest data rate, it is not suitable for any bandwidth hungry non-safety applications, as throughput of 64-QAM is the lowest among all the modulation schemes. The fact that the throughput does not decrease when the number of nodes increases is encouraging. This means that the throughput is insensitive to the number of nodes, and that the safety messages will be delivered without any problem even in dense network, and none of them will be lost if BPSK or QPSK is used.

Taking into consideration all the previous results, it can be affirmatively said that BPSK modulation with a data transfer rate at 3 Mbps is the most suitable modulation scheme for safety applications as judged by PDF, E2ED and throughput metrics.

4.1.4.2 Performance with respect to Message Intervals and with Number of Nodes

The previous section presented the performance of the messaging application with a periodic message broadcast interval of 0.05s for all four modulation schemes. These results affirm the suitability of the BPSK modulation for safety applications. The scope of this section is to identify the effect of varying broadcast message interval choice on safety application that utilizes BPSK modulation. Figure 4.6 to Figure 4.10 show the simulation results obtained for, respectively, total sent packets, total received packets, PDF, E2ED and throughput, against different number of nodes (x axis of the graph) ratio with five message intervals as a parameter.

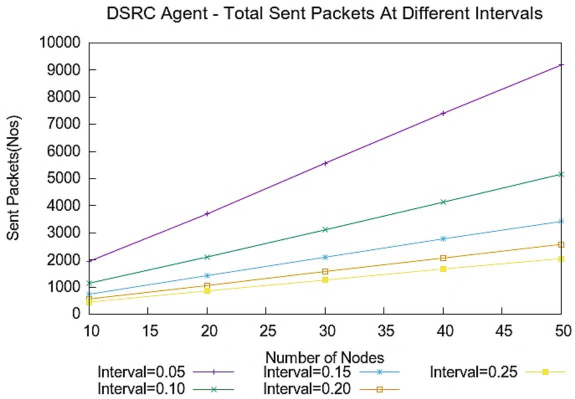


Figure 4.6: Sent packets at BSM intervals and No. of vehicles

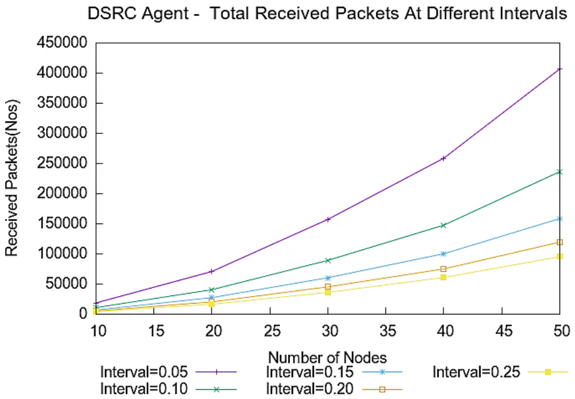


Figure 4.7: Received packets at BSM intervals and No. of vehicles

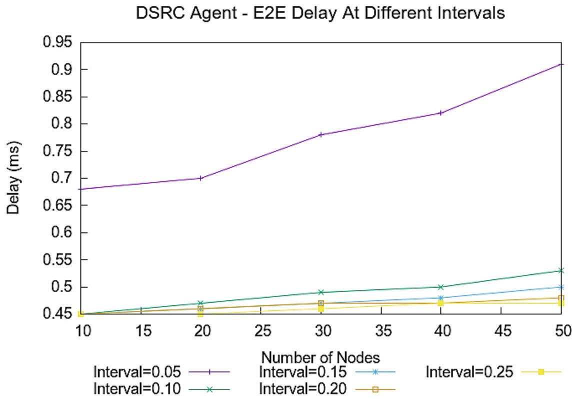


Figure 4.8: E2ED at BSM intervals and No. of vehicles

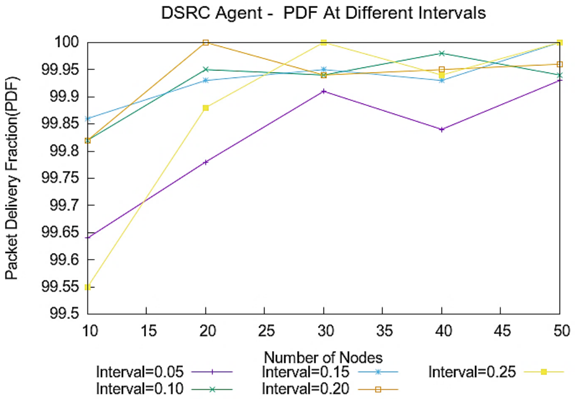


Figure 4.9: PDF at BSM intervals and No. of vehicles

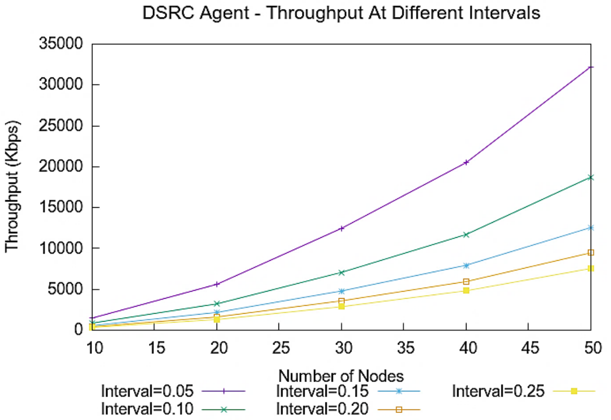


Figure 4.10: Average Throughput at BSM intervals and No. of vehicles

Comparing Figure 4.6 and Figure 4.7 in absolute numbers of sent and received packets, it can be seen that the number of received packets is considerably higher than that of the sent packets. This difference is expected as discussed in the previous section, wherein every vehicle is broadcasting message in one hop fashion to all neighbouring vehicles, so a vehicle is receiving more message than it is sending. The same figures show that as the messaging interval increases, the number of received packets decreases, which is an expected result too.

The E2ED performance results with variable message interval are presented in Figure 4.8. The result shows that as the message interval increases, the E2ED gets decreased. In addition, the delay is relatively stable as the number of nodes increase, ranging from 0.7ms to 0.9ms in the maximum case of the 0.05s message interval. In every other case, the E2ED is within acceptable limits for safety applications. As in the previous section, the fact that the E2ED never exceeds a limit reveals the real-time character of the system, allowing for in-time reactions of a driver, should there be an accident. Furthermore, it provides the necessary stability and reliability to the system, which is crucial especially in dense systems.

In Figure 4.9 the PDF performance results are presented. From this graph, it is seen that with every defined message interval rate the PDF is almost 99%. This result, along with results of Figure 4.6 and Figure 4.7, means that almost no safety message packet is lost when the number of vehicles increases. Thus, PDF rate with respect to message interval and BPSK modulation is in favour of the stability of the system, even in cases of growing vehicular density.

Finally, in Figure 4.10, the throughput performance with variable message intervals is presented. The throughput rate decreases as the variable message interval metric increases, but it increases with the increase of vehicular density.

Combining Figure 4.6 through Figure 4.10, it can be said that a message broadcast interval of 0.05s is the most optimum choice, as it provides the highest throughput at the cost of a relatively little high E2ED, which is considered to be within acceptable limits for generic time bounded safety applications. As in the previous section, the fact that the throughput does not get lowered when the number of vehicles increases, clearly, is a definitive indication of reliability and stability of the system, and it means that broadcast based safety data packets will not be affected when network density grows.

Another factor that influences the throughput performance is the mobility of the vehicles participating in the VANET system. Higher vehicular mobility causes lower throughput; this is because vehicular movement results in broken wireless links which takes time to recover and restart the transmission. The vehicular mobility is directly associated with BSM broadcast interval rate. In order to avoid re-transmissions, the BSM broadcast should adapt to the mobility of the nodes. In a real VANET environment, it is expected that the higher the mobility of the vehicle, the more the BSM broadcast.

4.1.5 Section Summary

In VANETs, of critical importance are the safety messages that are exchanged between the vehicles to meet the continuously growing safety demands. The aim of this part of the work was to identify the performance of two critical optimization parameters in safety applications in VANET systems, namely modulation scheme used for the transmission and time interval of broadcast message, along with the relation with the number of vehicles present in the system. The simulations were performed using NS2.35 with the appropriate mobility characteristics defined by the MobiSim simulator.

The overall results show that the use of a low index modulation scheme such as the BPSK modulation provides the best results, even as the number of vehicles in the system keeps increasing, with a very much acceptable E2ED level for BSM short applications. Furthermore, it is found that a relatively small broadcast message interval is in favour of throughput even when the number of nodes in the system is high, and this situation obtains at the expense of a little relatively high, yet acceptable, E2ED levels for safety applications.

4.2 A Study on the Performance of TCP based Application over DSRC based VANET

In this part of the research work, performance analysis of TCP based non-safety application is done, where integral periodic safety messaging application data traffic, instead of getting ignored, will continue to flow, using the most favourable BPSK modulation scheme at different message intervals.

4.2.1 Introduction

The design of DSRC is specifically meant for vehicular safety communication. The V2V and V2I communications, based on the IEEE 802.11p physical and MAC layers specifications, lay heavy emphasis on much sensitive and reliable short messaging application like vehicle collision avoidance application. However, the DSRC designers have envisioned possible integration of non-safety applications as well, using conventional transport layer TCP/UDP protocol. There is a possibility of using diverse applications that need reliable delivery of data using FTP for moving vehicles. As TCP was originally designed for wired networks, its use and behaviour do not mesh well with VANET systems. This contrast becomes all the more accentuated when safety messages coexist and periodically broadcast in short intervals of time.

Integral periodic broadcasting of BSM can never be ignored when using TCP types of communication over VANET, because safety services constitute the very core of the system. With regard to periodic BSM services, fair bandwidth sharing with TCP flows is an issue that needs careful examination, because IEEE 802.11p standard is designed without keeping the TCP fairness problems in mind. A big puzzling question in this regard is how much of DSRC bandwidth will be really available for a non-safety use, such as a TCP based infotainment type of applications, and whether it will be feasible at all to do such an IP like communications over 802.11p without affecting the reliability of existing BSM, which are already deployed for vehicular safety communication. These issues are addressed now and a simulation based study is carried out on the performance of TCP flows co-existing with DSRC safety messaging application over the same communication channel.

In this work, performance of TCP based file transfer application is studied with respect to BSMs broadcasting at varying time intervals. The effect of number of vehicles and their variable speed in the performance of protocol is not examined, as previous researchers have already done so, as mentioned in Chapter 2 (related work). The fact that 802.11p has been specially designed for safety application cannot be ignored even when analysing performance of non-safety application that operate using TCP agent. For this reason, it was decided to keep the integral BSM data traffic along with TCP flows. In general, safety short BSM payload size usually varies from 100 bytes to a maximum of 300 bytes, and here it is decided to keep 100 bytes as a typical size. Thus, during file transfer, beacon based BSM is periodically broadcasting high priority data traffic of 100 bytes.

4.2.2 Simulation Scenario

Scenario-2 and its associated parameters as described in section 3.9.2 and the four performance analysis metrics (throughput, PDF, E2ED and packet drop) are considered for the study of TCP performance. Based on these parameters, separate simulation is carried out for each TCP congestion control algorithm (Reno, Bic, Westwood, Highspeed, and Scalable). In this scenario, a few vehicles are accessing FTP resources from a RSU in multi-hop manner, while at the same time all the 50 vehicles are periodically broadcasting safety services at varying intervals. For broadcast based safety messages, no ACK in response is needed. All vehicular nodes are using the IEEE 802.11p MAC as presented in Chapter 2, and during an AIFS, each node (vehicle) must listen before sending data and back-off if the channel is or becomes busy. Since the data traffic is comprised of both broadcast safety messages and FTP over TCP, the BSM broadcast data traffic is assumed to occupy the bandwidth based on safety message interval rate, whereas TCP based traffic does so based on *cwnd* dynamics.

4.2.3 Simulation Results and Discussion

The following Figure 4.11 shows the *cwnd* dynamics of different TCP algorithms, while running non-safety messaging (TCP) file transfer application for a short interval of time (about 10s). The line chart shows performance in terms of *cwnd* versus elapsed time. It should be noted that the *cwnd* dynamics is not the aggregate result of multiple TCP flows but rather of only one TCP. The *cwnd* dynamics generated for MANET or

VANET is not straightforward to comprehend and is not generally reliable, because it does not reflect true protocol behaviour like saw tooth pattern obtained in wired networks. Despite this fact, the aim of the analysis and description made in following the line chart is to get a general idea in an abstract manner. So, arriving at any immediate conclusion about the protocol behaviour from this short duration of *cwnd* graph can never be definitive but can at best be only tentative. Here window size is increasing exponentially or linearly and there are drop-off events occurring randomly.

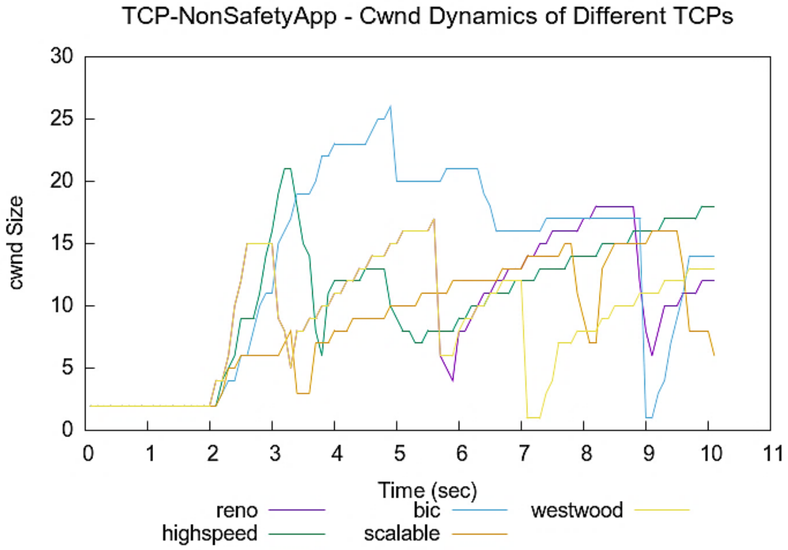


Figure 4.11: cwnd dynamic of TCP variants

The variation of the *cwnd* as time elapses is presented for each simulated TCP algorithm for the TCP file transfer application. The key difference among these TCP implementations lies in their *cwnd* growth behaviour in response to a congestion event, where packet loss results in reduction of the *cwnd*. In standard TCP, the value of the *cwnd* increases with each received ACK and exponential growth of the window size occurs at each RTT. Each TCP algorithm starts with the slow-start phase, which has more aggressive growth than the congestion avoidance phase.

As most of the TCP algorithms examined here have base characteristics of the TCP-Reno, the analysis starts with TCP-Reno. TCP-Reno has a change in its *cwnd* size, for the first 5 seconds by following exponential increase but with a drastic drop at about 6th sec, and after that it again goes up with exponential growth until 9th sec and thereafter it starts multiplicative decrease. The *cwnd* is continuously increasing as the ACKs are being received continuously until the 6th sec. Then, the packet loss occurs, which results in drastic decrease of the *cwnd* in order to compensate for the loss. After that, it starts growing again and the next loss occurs at the 9th sec, where the fast retransmit and fast recovery procedure starts over once again.

TCP-Westwood has performance similar to that of TCP-Reno. This behaviour is expected as they have a lot in common. The main difference is their reaction to the packet loss. TCP-Westwood also utilizes the fast retransmit and fast recovery to help recover more smoothly when packet loss occurs, as compared to TCP-Reno. This can be seen with *cwnd* event at 6th sec, where after the receipt of three consecutive duplicate ACKs or when timeout events occur, the TCP-sender resets the *cwnd* and *ssthresh* based on BWE. In comparison with other algorithms, the salient *cwnd* events can be seen at the 2nd and 7th second, where the *cwnd* growth is exponentially sharp. The entire *cwnd* phenomenon of TCP-Westwood reflects the characteristic behaviour of wireless channel, where channel error can cause congestion events.

The aggressive behaviour in the size of the *cwnd,* for the case of the TCP-BIC is obvious in this result: the length of the *cwnd* exponentially grows, taking advantage of the zeroth packet loss. After the receipt of triple duplicate ACKs, the *cwnd* length seems to be relatively squeezed at the 5th sec, as its length is adjusted, as expected, between the length of the no-loss window and the window with packet loss.

TCP-Highspeed is another algorithm that can be directly compared with TCP-Reno. The aggressive behaviour in *cwnd* growth is evident here too, but not as consistent as in the case of TCP-BIC. A more sharp increase is expected as the *cwnd* growth function of the TCP-Highspeed is higher compared to that TCP-Reno, as can be seen in the 2nd sec instant. In addition, when it comes to the case of packet loss, the behaviour of the two algorithms is, as expected, very similar, because they share the same *cwnd* decrease function. In addition, the performance is differentiated in favour of TCP-Highspeed when the *cwnd* is large enough. This is the reason for the consistent increase in performance from 6ths till 10ths.

Finally, TCP-Scalable has behaviour very similar to TCP-Reno. Even though it could not increase the *cwnd* very rapidly despite multiplicative increase characteristic, recovery is fast after a packet loss has occurred. It has the fastest recovery among all the algorithms presented, as it is independent of the *cwnd* length.

Figure 4.12 presents the RTT dynamics with the time elapsed for each TCP algorithm, with the TCP file transfer application lasting for 10 seconds of simulation. The RTT dynamics in the following figure is drawn for only one TCP flow; more explicitly, it is not an average of multiple TCP flows like with the other metrics used in this evaluation. Due to high vehicular mobility, the performance of TCP is affected in wireless environment. Vehicular mobility causes rapid changes in routing information resulting in long RTT and repeated time-outs. The level of congestion is also measured on the basis of sample RTT and the sender side *cwnd* size. From Figure 4.12, one finds that the RTT variation is in correspondence with the variation of the *cwnd*. The sender side estimates the current bandwidth consumed against every RTT [163].

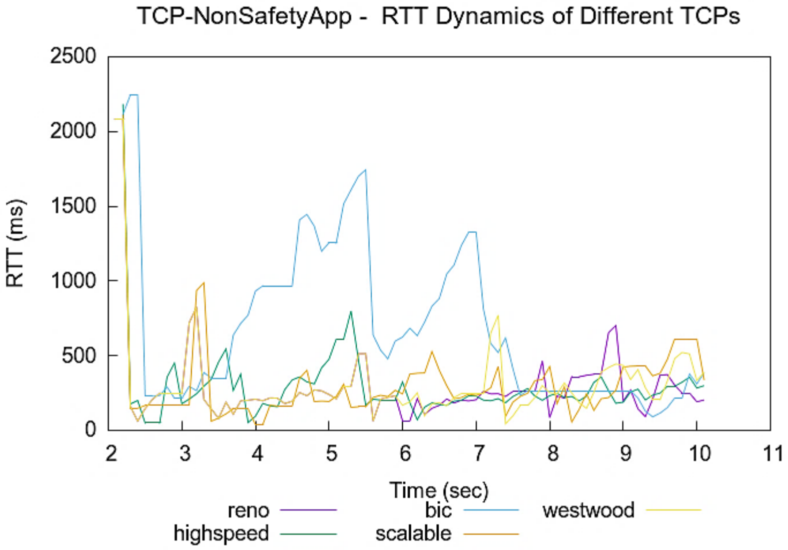


Figure 4.12: RTT dynamics of TCP variants

RTT metric also provides a way to measure the bandwidth allocation each TCP algorithm does, based on the data traffic flows in the network. In general, it can be said that the longer the RTT, the more the resources are consumed, resulting in poor performance in terms of throughput.

The two main measures used for the proper bandwidth allocation are: the slow-start characteristic and the congestion recovery algorithms. From the description of each algorithm in Chapter 2, it was shown that each one uses a slightly different kind of both slow-start and congestion recovery procedure.

As described earlier, TCP-Reno interprets time-out as an indication of serious congestion in the network, while at the same time it utilizes Fast Re-transmit and Fast

Recovery mechanisms to achieve more efficient transfer of packets in the network. These features bring out its good characteristics, as RTT is the lowest among the other TCP variants, and so TCP-Reno provides the best performance among all the other algorithms.

TCP-Westwood estimates the bandwidth by calculating the rate at which data are being delivered to the destination, which is the only main difference with TCP-Reno. This small difference is the cause for the main deviation compared to TCP-Reno and the initial bandwidth allocation took high RTT level at the slow-start phase. TCP-Highspeed is more aggressive than TCP-Reno even though it is very similar in its characteristics to TCP-Reno. Thus, very similar RTT performance is expected in the results. TCP-Scalable behaves identically to traditional TCP stacks when small windows are sufficient, as can be seen in the Figure 4.12.

TCP-BIC, on the other hand, with the use of binary search increase mechanism, allows TCP to occupy rapidly available network bandwidth. This aggressive behaviour results in a relatively poor performance with higher RTTs due to increased congestion.

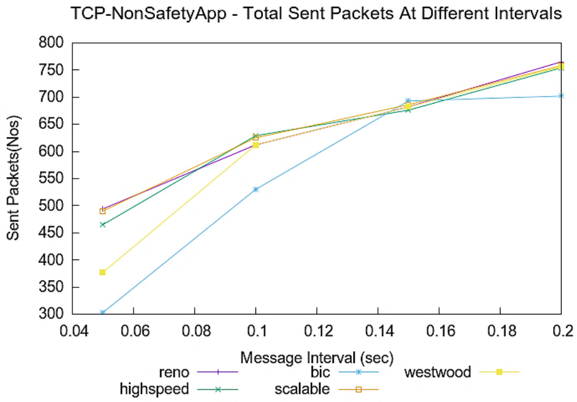


Figure 4.13: Sent Packets of TCP variants at BSM intervals

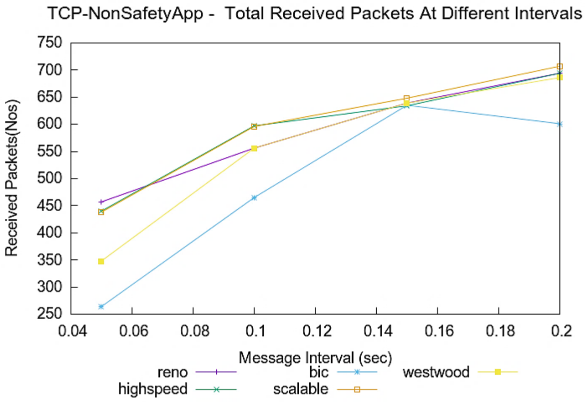


Figure 4.14: Received Packets of TCP variants at BSM intervals

The Figure 4.13 shows the performance in terms of total sent packets achieved by the non-safety messaging (TCP) file transfer application with respect to varying BSM intervals. As seen in this graph, the performance increases with respect to increase of BSM intervals. At the interval of 0.05s, the number of sent packets is very low, and TCP-BIC performance is at the lowest in contrast to other algorithms. Therefore, at lower interval, its performance is poor but at higher intervals of BSM traffic, TCP was able to send more data. The TCP-Reno, TCP- Scalable and TCP-Highspeed all show better performance at lower intervals.

Figure 4.14 shows the performance in terms of total received packets achieved by the non-safety messaging (TCP) file transfer application with respect to different BSM intervals rate. As seen in Figure 4.14, the performance of TCP is increasing with respect to increase in BSM intervals. At lower intervals, the performance is poor but at higher BSM intervals, TCP was able to receive more data. The reason behind good performance at high BSM intervals is due to availability of free bandwidth for non-safety use, which can be understood from bandwidth occupancy projection from Figure 1.2 of Chapter 1. The TCP-Reno, TCP-Scalable and TCP-Highspeed all provide better performance at lower intervals.

The following Figure 4.15 shows the performance in terms of average E2ED achieved by the non-safety messaging (TCP) file transfer application with respect to varying BSM broadcast intervals rate. E2ED measures the average time taken for a data packet transmitted from the source vehicle to the destination vehicles or RSU. E2ED of different algorithms are not easily interpretable from this line graph, as there are abnormal trends right from 0.05s till 0.2s of BSM broadcast intervals. In general, the E2ED in wireless channel is caused by re-routing delay, re-transmission, queuing delay, multi-path propagation delay and contention delay.

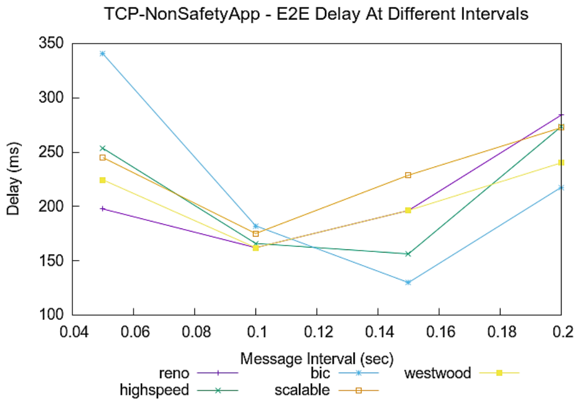


Figure 4.15: E2ED of TCP variants at BSM intervals

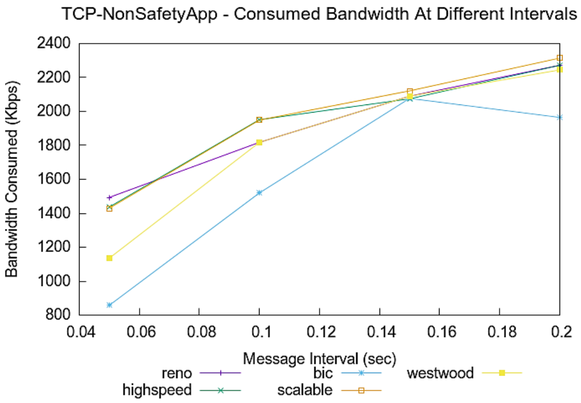


Figure 4.16: Average Throughput of TCP variants at BSM intervals

The results presented in section 4.1 shows that the overall BSM interval rate seems to be optimum at 0.05s and 0.1s, with the same 100 bytes of payload size. However, it would be of great interest to examine the performance of the TCP algorithms with respect to variable BSM interval rate, because BSM containing the same payload of the safety messages has different effects at variable broadcast interval rate.

The selection criterion for optimum algorithm is minimum E2ED, because low latency provides robustness to the TCP algorithm to congestion events and it can change its characteristics faster to deal with diverse applications that depend on low E2ED. The presented results can be better divided into two parts: the area where the E2ED is at a) optimum levels at 0.1s and 0.15s and b) non-optimum levels which occur at 0.05s and the 0.2s.

The deciding factor that changes through the variation of BSM interval rate and that causes variation of E2ED is the available bandwidth of the system. The larger the BSM broadcast interval rate, the more available the bandwidth becomes for TCP use. Thus, E2ED figure captures the performance of each algorithm to changes in the bandwidth.

Again, the basis of the analysis will be TCP-Reno because it is the baseline algorithm, and the goal is to analyse the comparative performance. The results show that a significant performance improvement is achieved for 0.1s and 0.15s of message interval cases, whereas for 0.19s case there is decrease in the performance of all the algorithms. Very small message intervals lead to inaccurate estimation of the available bandwidth, because of more TCP packet losses. Medium BSM message interval rate is adequate for a more precise bandwidth estimation, and for making better and fair utilization of bandwidth possible. An advantage of TCP-Westwood compared to TCP-Reno is the better bandwidth estimation at 0.05s of message interval rate. The outcome of this is better bandwidth utilization, less lost packets and consequently, smaller E2ED.

TCP-Scalable and TCP-Highspeed appear to have very similar behaviour at 0.05s, as they share the same properties. As the message interval rate increases, the bandwidth utilization for TCP becomes better and the E2ED decreases. Even though it seems that when the message interval lengths are too large, the results are not good as the E2ED gets higher for TCP-Scalable. Finally, the aggressive behaviour of TCP-BIC is obvious in the highest E2ED among all algorithms at the interval of 0.05s. These results reveal a very good performance for the TCP-BIC and TCP-Westwood especially at 0.15s, which appears to be a good trade-off between bandwidth utilization and E2ED.

As explained earlier, the variation of broadcast message interval rate resulting in the variation of the E2ED can be used as a metric to gauge available bandwidth of the system. In addition, the available bandwidth of the system can be estimated from the ACK that the sender receives back. Timely and consistent receipt of ACK in VANET is a challenging task, because of several issues, such as path loss, BER and frequent RTO. A VANET system with low vehicular mobility has higher chance of successfully receiving ACKs on time because of less channel error, whereas fast moving vehicle loses more ACKs. Thus, it is obvious that a system with a lower mobility gets a better BWE. However, low BSM interval rate causes more TCP re-transmission. Nevertheless, as can be seen in Figure 4.15, these two facts conveniently conspire to produce better E2ED results in the medium values of the BSM interval rate, rather than lower or higher rate,

Figure 4.16 shows the performance in terms of average consumed bandwidth of TCP variants with respect to BSM broadcast intervals rate. Average consumed bandwidth measures the total number of packets successfully delivered to destinations over a period of time. It is evident from the line chart that as the BSM broadcast interval rate increases, the TCP throughput also increases. It is because the BSM traffic, which is coexisting with the non-safety FTP application, delays the broadcasting time in accordance with the message interval time. The broadcasting delay is highest in the line chart with the highest interval time of 0.2s. This broadcasting delay frees the BSM traffic for that amount of time and within this slot the TCP based application fully utilizes the available bandwidth, and as a result, with the increase of message interval time, the TCP throughput also increases.

Varying vehicular density has not been considered in throughput and other metric evaluation because it is a fact that as the density of vehicle increases, the common wireless channel access contention increases too. The channel contention, frequent route disconnection and re-routing cause delay in intermediate vehicles; as a result, RTO makes TCP sender trigger frequent slow-start which degrades the throughput performance of all the TCP variants.

Figure 4.16 shows the line chart where the TCP-Scalable and TCP-High-speed exhibit best performance among all the TCP algorithms. It is evident that the safety messaging application intervals impact the TCP throughput. The TCP-Reno, TCP-Scalable, TCP-

Highspeed and TCP-Westwood have very similar performance at the message interval of 0.15s. However, TCP-BIC is different due to its aggressive strategy, and its performance is the lowest among all the algorithms at every message interval except at 0.15s.

Another factor that influences the throughput performance is the mobility of the vehicles that the VANET system is based on. Higher vehicular mobility causes lower throughput, because vehicular movement results in frequent broken wireless links, and it takes time to recover and restart the transmission. The vehicular mobility is directly associated with BSM broadcast interval rate. In order to avoid re-transmissions, the BSM interval is usually adapted to the mobility of the nodes. The higher the mobility of the vehicle, the more BSM is broadcast. However, it must be noted that the TCP congestion control algorithms cannot distinguish the difference between the need for re-transmission that comes from an ACK loss due to higher mobility and an ACK loss that comes from buffer overflow. All the TCP congestion control algorithms manage to have a very similar performance at the message interval rate of 0.15s while dealing with mobility of vehicles.

The E2ED results presented in Figure 4.15 may be viewed in relation to the throughput performance results presented in Figure 4.16. The E2ED is directly associated with the bandwidth utilization and better bandwidth utilization results in higher throughput. As the end-to-end latency decreases, it is obvious that the respective consumed bandwidth increases. This is expected as the VANET system is able to send more packets in less time and to increase this way the average throughput. Furthermore, the throughput is also heavily dependent on BSM broadcast interval rate rather than on only E2ED, as a small increase in the E2ED does not always mean that the throughput will drastically decrease too.

From the results of Figure 4.15 and Figure 4.16, it can be said that the best choice would be the TCP-Highspeed at 0.15s of message broadcast interval rate, because it is this choice that provides the best trade-off between the lowest E2ED and peak throughput.

Figure 4.17 shows the performance in terms of dropped packets of non-safety messaging (TCP) file transfer application with respect to different safety message interval rates. The data packets drop metric is complementary to the PDF, where the number of dropped packets decreases when the PDF increases and *vice versa*. Higher vehicular mobility, wireless channel error and buffer overflows at the intermediate vehicles are attributed to packet losses.

As shown in this line chart, TCP-Scalable and TCP-Highspeed show better performance overall than other algorithms.

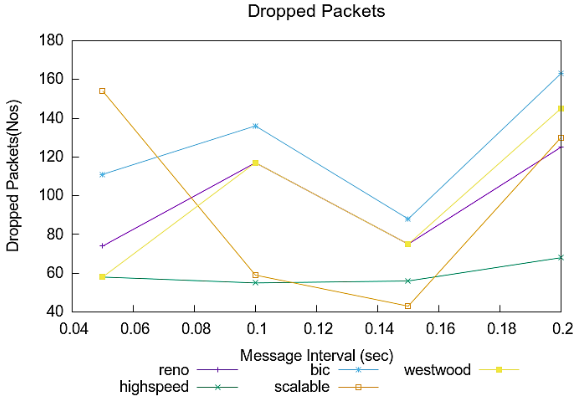


Figure 4.17: Total dropped packet at different BSM intervals

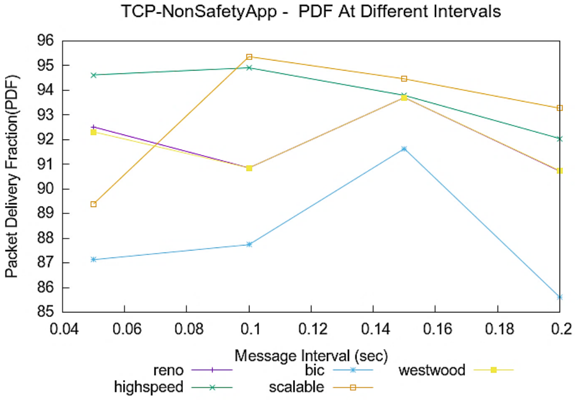


Figure 4.18: PDF of TCP variants at different BSM intervals

Comparing PDF and packet dropped results, the characteristic duality becomes obvious in every case. However, in some congestion algorithm cases, the number of dropped packets seems much higher and this may look like going against the previous observation. However, these are absolute numbers and what really matters is the PDF and not this absolute number. As stated above, the number of dropped packets is complementary to the PDF.

Figure 4.18 shows the highly important performance criterion *viz.* PDF, achieved by the non-safety messaging (TCP) file transfer application with respect to different BSM intervals rate. The higher the PDF rate, the better the performance of an algorithm. Furthermore, an important criterion to be kept in mind is the stability of the PDF delivery. For example, an algorithm with a slightly lower but constant (i.e. smaller variance) PDF is generally preferable to the one with a PDF that is slightly higher by the same extent but with larger variance.

TCP-BIC, like the previously presented results, behaves differently compared to the rest of the TCP algorithms due to its aggressive nature. By being so, it touches the lowest achievable PDF along with a high variance in its values. Thus, it becomes unreliable and consequently unsuitable for VANETs.

The rest of the algorithms examined have somewhat similar performance without having any significant difference in trends. They are more or less stable in the variation of the BSM broadcast interval rate. PDF with a drop or increase of 2% is considered marginal and generally inconsequential in the context of FTP based applications, and the performance an algorithm with such variation can be considered as stable. The ones with the best performance among all the algorithms are the TCP-Highspeed and TCP-Scalable, as they achieve somewhat consistent behaviour in all the BSM interval rates.

4.2.4 Section Summary

There are some requirements to be met if one should have TCP dependent applications like FTP in VANET for various infotainment purposes. In the case of DSRC based VANET, the integral BSM periodic broadcast has to necessarily coexist with the TCP transmission. The high priority beacon based BSM does have severe negative effect on the TCP flow but at no time can it ever be ignored. In section 4.2.3, performance analysis and comparison were conducted on five different TCP variants (TCP-Reno, TCP-BIC, TCP-Westwood, TCP-Highspeed and TCP-Scalable) in order to evaluate them and decide on their suitability for implementation in VANET systems. In addition, variable BSM broadcast interval rate was used in order to examine its impact on the system's performance for all the TCP algorithm cases. The results obtained are in accordance with the characteristics of each algorithm such as the bandwidth exploitation, the bandwidth change, and congestion incident etc. From these results, it can be said that the algorithms with the most consistent and acceptable behaviour are the TCP-Highspeed and TCP-Scalable, as they exhibit favourable behaviour and features such as high and stable PDF, low packets drop along with comparatively increased throughput.

4.3 A Study on the Impact of TCP Flow over Periodic BSM

4.3.1 Introduction

If one tries to reclaim the unused bandwidth of DSRC channel for low priority non-safety application over the much reliable modulation (BPSK) scheme, it will certainly have negative impact on the BSM flows. From the perspective of VANET, the non-safety services are low priority data flow, which should only be considered if the bandwidth permits. As envisaged in the DSRC architecture, it is highly unlikely that in future full-fledged protocol deployment, only the beacon based safety data traffic will be used exclusively in VANET. Therefore, sooner than later, non-safety services are also very likely to become necessary and unavoidable.

There are some services such as toll tax collection, road side service information and congestion charging etc., which are some of the important non-safety use cases required for vehicles in motion. While using such applications, safety applications, that are integral to any VANET system, cannot and should not be ignored because of its necessity in preventing accident. Beacon based time sensitive BSM traffic, which periodically broadcasts in the VANET system, occupies a major part of the bandwidth, while unused part of bandwidth can be better utilized for infotainment purposes.

Deployment of non-safety application over highly congested network entails severe consequences on the performance of high priority BSM traffic flow. Though it is not yet known how exactly the impact of traffic changes protocol behaviour and under what traffic conditions it happens, it is necessary to examine the extent of impact of TCP based time-insensitive low priority traffic on delay-sensitive safety services.

FTP based flows are often regarded as long lived reliable data traffic that can occupy the bandwidth left unused by high priority safety data traffic. The non-safety data traffic negatively impacts on the performance of delay-sensitive safety applications. In the presence of time-sensitive traffic, FTP too suffers from its inefficiency in upload or download strategy which comes from peers or RSUs.

There are many reasons why the inclusion of non-safety data traffic coexisting with safety flow, should be considered an important criterion to gauge the performance of DSRC protocol testing. The characteristic of such low priority traffic can cause a range of dynamic behaviour, such as varying packet loss pattern, variation in packet queue, packet length as well as bandwidth link utilization at bottleneck spots, and these can have significant effect on the performance of BSM. A VANET due to its dynamic topological environment may cause randomness in packet arrival associated with delay.

The performance evaluation of DSRC protocol with no realistic high priority broadcast based traffic coexisting with delay insensitive traffic as studied by many researchers (discussed in Chapter 2, literature review section), does not contribute to realistic and comprehensive protocol investigation. So, in this work, the performance of BSM application is examined when there is low priority traffic generated from TCP congestion control algorithms. The BSM application broadcast interval rate is turned into a variable in order to identify the effect of bandwidth occupancy from the TCP traffic. Furthermore, as each TCP algorithm has a different way to predict the bandwidth utilization and changes according to its characteristics, widely used loss based TCP congestion control algorithms are utilized in order to study the system's performance. In addition, the case of WithoutTCP traffic flow is also included to compare the effect of TCP traffic on the system's performance.

4.3.2 Simulation Scenario

Scenario-2 and its associated parameters as described in section 3.9.2 and the four performance analysis metrics (throughput, PDF and E2ED) are taken up for the study of impact of TCP over BSM. Based on these parameters, each TCP congestion control algorithm (Reno, Bic, Westwood, Highspeed, and Scalable) has been simulated separately. A few vehicles in the scenario access long-lived FTP resources from RSU in a multi-hop fashion, while at the same time all the 50 vehicles are periodically broadcasting high priority safety services at varying intervals.

This scenario has integral beacon based BSM, broadcasting at varying intervals; therefore, no ACK in response is expected. All vehicular nodes are using the MAC method of IEEE 802.11p which is based on CSMA/CA, as presented in Chapter 2. Since the data traffic is comprised of both beacon based safety messages and FTP over TCP that are coexisting over same channel, the BSM broadcast data traffic is assumed to occupy the bandwidth based on BSM interval rate, whereas TCP traffic is based on *cwnd* dynamics.

4.3.3 Simulation Results and Discussion

Figure 4.19 to Figure 4.23 present the simulation results of BSM application in the presence of different TCP algorithm with respect to total sent packets, total received packets, PDF, E2ED and throughput ratio respectively, with the different broadcast message interval rate.

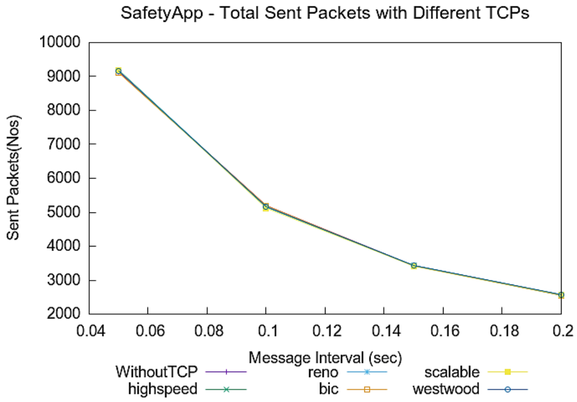


Figure 4.19: BSM Sent Packets in the presence of TCP variants

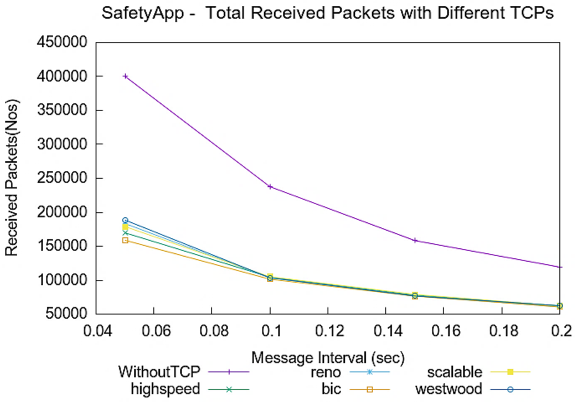


Figure 4.20: BSM Received Packets in the presence of TCP variants

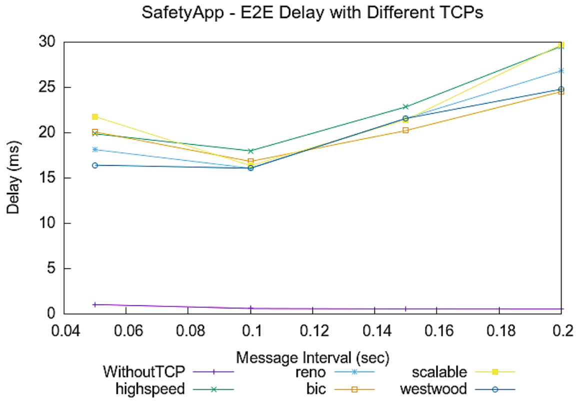


Figure 4.21: E2ED of BSM in the presence of TCP variants

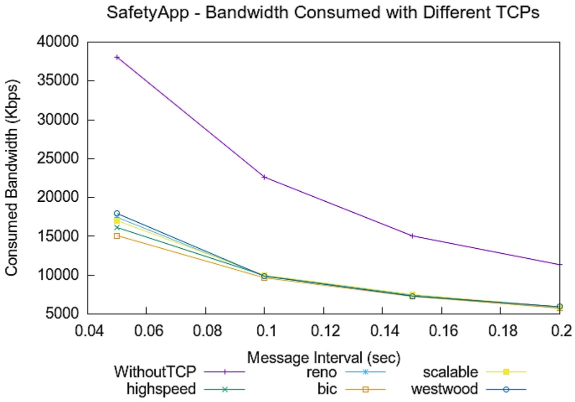


Figure 4.22: Average Throughput of BSM in the presence of TCP variants

Figure 4.19 shows the performance in terms of total sent packets achieved by the BSM at different broadcast intervals rate and with different TCP flows. As shown in this graph, the send performance of the messaging application is not at all affected by any kind of TCP flows, but as the broadcast interval increases, the packets received by all the algorithm decrease. The reason for this effect is obvious: high broadcast interval reduces the sending rate.

The line chart in Figure 4.20 shows the performance in terms of total received packets achieved by the BSM at different broadcast intervals rate and with different TCP algorithm flows. As shown in this graph, the received packet count was affected by all kinds of TCP flows, and the effect is considerably high at higher broadcast interval rate of the safety application.

It is obvious that when there is no TCP flow (WithoutTCP case) in the background, the BSM application can make use of the whole channel bandwidth, and in this way the number of the received packets is based on the message broadcast interval rate in comparison to the cases with TCP flow. In these cases, the behaviour to be examined is how BSM broadcast interval rate influences and impacts the availability of the channel bandwidth and the ability of low priority TCP traffic that tries to jostle for any available bandwidth.

When the BSM broadcast interval is low, the channel bandwidth is temporarily and relatively free for that interval of time. At this point of time, the TCP which is running at low priority, seizes the moment to occupy any leftover available bandwidth. From Figure 4.20, it can be seen that the most interesting trend occurs at 0.05s of message interval rate, whereas in the rest of the cases, the performance of BSM remain almost the same. From 0.05s till 0.08s, the varying effect on BSM is due to the impact of convergence time and *cwnd* growth taken by each TCP algorithm. The uniform performance from 0.1s till 0.2s comes from the fact that the system has reached a stable state, in the presence of low priority transmissions; thus, the free bandwidth space remains nearly the same, and as a consequence the impact of all TCP algorithms on BSM has the same performance.

The criterion to be considered here is better bandwidth estimation made by TCP flow. The less aggressive the available bandwidth estimation, the more the transmitted packets are possible. TCP-Reno, TCP-Westwood and TCP-Scalable have almost similar performance as their bandwidth estimation procedure is very similar, whereas the aggressive bandwidth estimation of TCP-BIC and TCP-Highspeed results in less BSM received packets.

Figure 4.21 shows the performance in terms of E2ED achieved by the safety messaging application at different broadcast intervals rate and with different TCP flows. As shown in this graph, the E2ED was affected by all kinds of TCP flows, and it was considerably much affected at higher broadcast interval rate. As seen in this graph, TCP-Scalable and TCP-Highspeed suffer less than other TCPs.

It is obvious that the lower the E2ED, the better the performance of the BSM application, as it exhibits full bandwidth utilization. The differences in bandwidth utilization between WithoutTCP and with TCP cases is clear in Figure 4.20. When the channel is occupied by WithoutTCP (BSM flow alone), the whole bandwidth is available for the BSM application. Thus, BSM makes full use of bandwidth and the lowest degree of some latency in transmission is due to wireless channel and distance between communicating vehicles. The presented results can be divided into two parts: the range where the E2ED is at acceptable levels, which is at 0.05s and 0.1s cases, and the other, at non-acceptable levels at 0.15s and 0.2s cases.

The TCP-Highspeed, TCP-BIC and TCP-Scalable algorithms exhibit more aggressive behaviour; so, they occupied bigger portion of the available bandwidth. This is reflected as lower bandwidth utilization from the BSM application, and as a consequence, leads to high E2ED. On the contrary, TCP-Reno and TCP-Westwood, which do not show aggressive behaviour, occupied the bandwidth faster and as a result, the latency is not comparatively higher.

Figure 4.22 shows the performance in terms of consumed bandwidth (throughput) achieved by the safety messaging application at different broadcast intervals rate and with different TCP flows. As shown in this graph, the consumed bandwidth was affected by all kinds of TCP flows, and it was considerably so at higher broadcast interval rate of the safety application.

Comparing the results presented in Figure 4.20 and Figure 4.22, it is evident that the correlation between the number of received packets and the throughput is obvious. The higher the received packets, the better the throughput of BSM application, as expected. The bandwidth utilization for safety application depends upon the BSM broadcast rate; it is the most critical performance impact factor. When there is no TCP flow, all the available bandwidth is occupied by the high priority BSM traffic. But when TCP application traffic sets in, the BSM performance gets affected, despite being high priority traffic and this is evident from Figure 4.22. The reason behind this is once the channel is occupied by TCP traffic, it continues to access the channel based on the allocated time slot. Thus, BSM obtains considerably lower throughput levels in comparison with WithoutTCP flow.

As can be seen, the higher the broadcast interval rates, the lower the throughput of the system. From the perspective of consumed bandwidth, the most favourable case is at 0.05s of BSM message interval rate. When the BSM message interval is high, fewer messages are sent, resulting in smaller average throughput, as is expected. But when the message interval is small, more packets are sent and thus a high BSM throughput is obtained.

The average throughput performance with respect to impact of TCP algorithm seems to be quite similar in all the cases. There are some small discrepancies at the message interval rate of 0.05s, though in the rest of the cases, the behaviour is similar. The small discrepancies at 0.05s case arise from the fact that the BSM occupies the channel bandwidth based on the characteristic of each TCP algorithm. It can, therefore, be concluded from the consumed bandwidth that the throughput result is impacted by all the TCP flows, and at higher BSM broadcast interval rate, it is indeed affected very much.

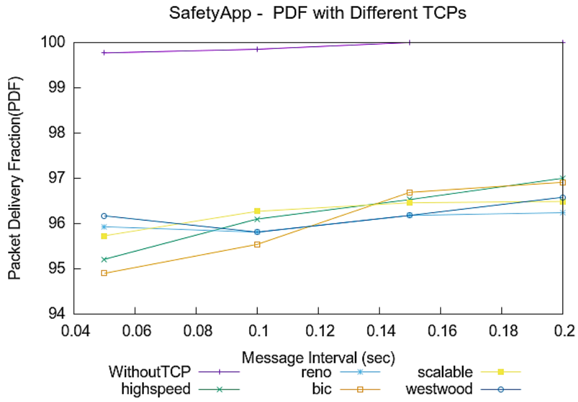


Figure 4.23: PDF of BSM in the presence of TCP variants

Figure 4.23 shows the performance in terms of PDF achieved by the safety messaging application at different broadcast intervals rate and with different TCP flows. PDF is also a very important parameter to gauge the performance of the system or use case. The higher the PDF, the better the performance of the application. Another related, important factor that reflects the good performance of the system in terms of PDF is its low variance throughout the transmission duration. As shown in Figure 4.23, the PDF is affected by all kinds of TCP traffic, the variation being a little larger at lower end than at higher end of the broadcast interval rate.

Similar to all the previous cases, the performance of PDF with and without the use of TCP flow case is evident. When there is no low priority TCP traffic (WithoutTCP case) in the background, the PDF of BSM is excellent because bandwidth is solely occupied by the safety data traffic. In the rest of the cases, the under-performance of BSMs, as shown in the PDF line chart curve, indicates the impact of bandwidth being occupied by the background TCP traffic.

As the BSM broadcast traffic goes high (0.05s interval rate), the network bandwidth gets heavily congested with safety data traffic. In this congested state, the TCP flow at the same time tries to seize or grab every opportunity to exchange FTP based non-safety resources from RSU, which causes frequent invocation of TCP slow-start algorithm. This results in unfair sharing of bandwidth, whose negative impact on the BSM delivery ratio is evident.

The variance of the impact of all the examined TCP algorithm cases seems to be marginal in comparison with WithoutTCP case. But in terms of reliable delivery of time critical BSM application, this variance is not easily acceptable.

In conclusion, it can be said that the PDF of BSM is affected by all kinds of TCP in contrast to WithoutTCP flow and it is considerably so at lower broadcast interval rates. TCP-BIC and TCP-Highspeed show comparatively higher impact than other TCP flows.

4.3.4 Section Summary

In this section the impact of low priority TCP traffic over high priority BSM flow is examined carefully. The aim is to identify the degree of impact by TCP flow on the performance of BSM applications. The most commonly used TCP congestion control algorithms such as TCP-Reno, TCP-Scalable, TCP-Highspeed, TCP-BIC and TCP-Westwood are employed for this purpose. Their basic difference lies in the way they adjust *cwnd* and the channel bandwidth they occupy based on the bandwidth estimation. The bandwidth occupancy or utilization is primarily determined by BSM broadcast interval rate and the characteristics each used TCP algorithm exhibits.

In contrast to WithoutTCP, the impact of TCP algorithm over BSM shows severe performance deterioration, which is not as expected according to the discussion made in relation to Figure 1.2. The performance deterioration is more severe in cases of TCP-BIC and TCP-Highspeed, because they aggressively occupy their bandwidth. Finally, it is observed that using a few TCP flows over congested VANET system takes its toll on the performance of high priority integral BSM applications. Therefore, it becomes imperative that impact on high priority BSM traffic must be taken into serious account, while designing future applications that rely upon TCP agents.

4.4 Chapter Summary

In this chapter, an incisive study has been made to understand the real characteristics of BSM application and TCP based non-safety application in a freeway environment with sparse vehicular density and high vehicular speed.

The results obtained in the first part reveal that the low index BPSK modulation scheme is in favour of safety applications that require one hop periodic broadcasting. The performance of BSPK modulation scheme in terms of throughput and PDF is unaffected with respect to growing number of vehicles. As BSM broadcast interval rate depends upon the severity of dangerous traffic situation, we evaluated the optimum message broadcast interval rate over the most suitable BPSK modulation scheme. The simulation result suggests that the optimum message broadcast interval rate for a generic time-sensitive application falls within 0.05 – 0.1s.

Growing demand for non-safety infotainment applications necessitates that the possibility of its deployment over DSRC protocol be tested carefully and thoroughly. As a test case, in the second section of this chapter, the performance of an archetypal TCP application *viz.* FTP application is studied, while concurrently running the default delay-sensitive safety application over BPSK modulation scheme and at varying message interval rates. It is found that TCP based application when coexisting with beacon based sensitive applications indeed creates severe network congestion. In view of this, the behaviour of five congestion control variants of TCP (TCP-Reno, TCP-BIC, TCP-Westwood, TCP-Highspeed and TCP-Scalable) are evaluated. The results obtained show clearly that the TCP-Scalable and TCP-HighSpeed have relatively better performance in terms of PDF, throughput and delay.

The DSRC protocol architecture permits a mix of transport layer flow like TCP in parallel with WSMP but with higher priority of access to BSM application. When, for practical reasons, one is constrained to using single radio devices, it becomes necessary to investigate the effect of using TCP based delay-tolerant application over default BSM application flow. So, in view of this, in the third part, we studied the impact of TCP over BSM application. The simulation results demonstrate clearly that there is serious impact of a few TCP flows on BSM application.

Performance of TCP based application and its apparently unavoidable negative effect on BSM application clearly show that the scope of deployment of non-safety application over DSRC protocol is severely limited. This is fundamentally due to the obtrusive and interfering nature of TCP, when used in parallel with high priority BSM applications.

To circumvent this difficulty and to overcome this drawback, it is imperative that the *unruly* or aggressive behaviour of TCP application is somehow *tamed* or contained. The strategy this thesis suggests to achieve this is to control the congestion control algorithm by adaptively tuning or freezing its slow-start mechanism. The following two chapters, which constitute the main contribution of this research, explicate the implications and describe the implementation of this innovative proposal on two standard congestion control mechanisms *viz.* TCP-Sclable and TCP-LP, along with an assessment of its success and an appraisal of its suitability in the context of the principal problem addressed in this work i.e. co-existence of TCP applications with BSM safety messages.

Chapter 5

vScalable-TCP: An Optimized TCP-Scalable Algorithm for Sharing Bandwidth Cooperatively with Safety Applications

5.1 Introduction

VANET communications based on the IEEE 802.11p physical and MAC layer specifications tends to put heavy emphasis, as is to be only expected, on much-sensitive and reliable short messaging applications. At present, there exist a variety of important non-safety services and entertainment types of applications that are required to be deployed over VANET. Transport layer protocol TCP is commonly used to cater to such reliable delivery of services. But the earlier simulation studies show that the use of non-safety applications that rely on TCP adversely affects the performance of core DSRC's safety messaging applications. Further, the requirement specifications and performance analysis results indicate that despite having standard TCP in DSRCs protocol stack, its usefulness in VANET is very limited and hence its deployment remains severely circumscribed.

Performance analysis in section 4.1 showed that DSRC's channel bandwidth occupancy is linked to the message interval rate, wherein broadcasting interval rate is directly proportional to the channel occupancy. When distance between the vehicles is close and the speed is high, it is obvious safety message generation rate should be high. A safety system with high message generation rate provides sufficient time to accident prone vehicles in taking appropriate timely action to prevent accident either by actuating safety sensor or by means of alert to driver. The previous simulation study suggests and informs that safety message generation at the rate of 0.05s or 0.10s is optimum for generic delay-sensitive and high priority safety applications.

Beacon based periodic broadcast of BSM, when used singly at the message interval rate of 0.05s or 0.10s or used adaptively, causes variable amount of bandwidth wastage. Of course, under the current DSRC protocol draft specification, the unutilized bandwidth can be made available for non-safety use such as TCP based file transfer applications. Theoretically speaking, the limitation of TCP over congested DSRC channel arises on account of BSM broadcast rate and TCP's inherent congestion control mechanism. So, in order to explore the possibility of deploying non-safety application under DSRC protocol and to estimate to what extent this is feasible, another simulation based performance analysis study of FTP over TCP is undertaken in section 4.2.

Standard and high speed congestion control TCP variants are considered in section 4.2 to assess the performance of individual algorithms. Since the algorithm of each TCP variant employs its own unique method to handle the congestion in the network, the simulation results of TCP-Reno, TCP-BIC, TCP-Scalable, TCP-HighSpeed and TCP-Westwood showed performance discrepancies in terms of throughput, packets dropped and E2ED metrics at different safety message generation rates, which is obvious. Overall performance evaluation showed that TCP under performs in the VANET scenarios studied here due to several reasons such as frequent link failure, buffer overflow, slow convergence, routing overhead, and of course, the coexistence of high priority integral broadcast traffic.

When a few FTP services over DSRC try to occupy unutilized channel bandwidth, it severely undermines the performance of BSM which was not as expected. This is evident in the study of performance analysis of impact of TCP on DSRC in section 4.3. The reason for performance degradation comes from unfair channel occupancy from TCP, channel contention, routing overhead, route failure and propagation delay. The reason for unfair channel occupancy, in turn, is due to frequent calls to slow-start mechanism in the event of congestion. Also, while TCP with large difference in the

RTT and packet drop interval rate allows good flow of heterogeneous time-sensitive traffic, TCP connection with consistently smaller RTT along with smaller packet drop interval rate does not favour BSM flows. This means that TCP with smaller RTT or TCP making frequent calls to slow-start mechanism tends to grab higher share of wireless channel, thereby hurting unfairly the critical periodic safety applications.

When periodic BSM services and TCP based non-safety applications are brought together over the same channel, serious questions arise as to how they are going to share the available bandwidth and this is a matter of grave concern for safety applications. Using unutilized bandwidth in a fair and just manner is a tough ongoing challenge in standard TCP design. Fairness in TCP flows under DSRC based VANET is an issue that needs careful study because IEEE 802.11p standards are not designed with the TCP's bandwidth distribution problems in mind.

Fair distribution of non-safety data traffic using TCP agent over 802.11p, without much affecting the reliability of existing BSM applications, is the main challenge that is addressed upfront in this study. Towards this end, the TCP-Scalable [75] variant is studied and a novel vScalable-TCP algorithm is designed and optimized for dedicated use in VANET. The proposed vScalable-TCP congestion control mechanism is found to be suitable for basic uses of non-safety applications, while at the same time not interfering much with safety messaging applications.

5.2 Problem Description

Drivers and passengers nowadays take for granted the availability and desirability of utility services like electronic toll collection, and infotainment related services form Internet that require TCP/IP connection from RSU or other moving vehicles. Use of TCP types of data flow in VANET environment causes network congestion, which arises from poor congestion control mechanism *wired* into traditional TCP. In several earlier works, different types of TCP congestion control algorithms were studied and evaluated as to how they perform in dynamic VANET environment. TCP performs well in fixed wired and static wireless topology; however, due to high vehicular speed, on-board wireless communication technology inevitably brings in discrepancies in transmission quality and degradation in performance. Previous investigation in [108] showed that TCP provides poor bandwidth consumption in *ad hoc* networks; however, it was surmised, hypothetically, that higher throughput was possible and can be achieved. Performance degradation of TCP in VANET environment is caused mainly by conservative flow and congestion control mechanisms.

In general, packet loss in TCP can occur in two ways, either by receiving triple duplicate ACKs or by not receiving ACK in time. In the event of a packet loss due to congestion, TCP adjusts the congestion window by adaptively sliding it to prevent further packet loss. TCP slow-start and congestion avoidance mechanisms are usually used to control the optimal flow of data, and at the same time, prevent congestion in the network. In the standard TCP, the slow-start begins with sender window of size equivalent to 1 MSS. After successful receipt of each ACK the window size gets doubled from previous count. The exponential increment of congestion window continues until it reaches slow-start threshold. Once reached, it enters into congestion avoidance phase, where the window is increased additively until packet loss occurs. If the cause of the packet loss is from RTO expiry, then the slow-start again begins with *cwnd* of 1 MSS size. However, if the packet loss is detected by receiving triple duplicate ACK, then the algorithm enters into fast recovery state.

Deployment of transport layer protocol TCP over DSRC based VANET raises serious issues in controlling the desired optimal flow of data packets. As stated earlier, TCP in VANET assumes transmission errors come from congestion events, and as a result, it reduces the throughput by implementing slow-start and congestion avoidance mechanisms. TCP's congestion control mechanism is applied to optimize the throughput, to properly utilize the network bandwidth, and of course, to meet application requirements. However, the traditional TCP congestion handling procedure proved inadequate, inefficient and unsatisfactory in dynamic VANET environment.

There are three different ways by which TCP's inefficiency issue in VANET environment can be tackled. The foremost and perhaps the simplest option is to modify traditional TCP's congestion control mechanism. Secondly, one can utilize information from intermediate systems and finally, the least recommended one is to design an altogether new transport layer protocol that is not based on TCP at all but suitable for deployment of reliable infotainment application in VANET.

We have elected the option of tuning and tweaking the traditional TCP's congestion control mechanism to handle the inefficiency issue in VANET. The behaviour of TCP is simply controlled by two functions *ssthresh* and *cong_avoid*. There are some procedures to tweak traditional TCP's slow-start and congestion avoidance mechanisms in order to optimize TCP that is suitable for VANET. The present work proposes a unique approach to handle slow-start at the congested situation. In this novel method, at highly congested situation, the slow-start is suppressed temporarily to avoid further congestion. The main purpose of doing so is to accord more importance to the existing safety message flows, so that the TCP flows of non-safety application does not much harm or hamper high priority BSM applications.

5.3 Analysis & Implementation of TCP-Scalable on NS2 Linux TCP

Performance analysis of five different TCP algorithms have been analysed in section 4.2, out of which TCP-Scalable is chosen arbitrarily as a typical case for the optimization experiment. The main reason for doing so is that the proposed optimization mechanism is expected to be applicable and adaptable to any existing congestion control mechanism. As per RFC-2581 and RFC-5681 recommendation, the congestion window should be halved for each packet loss in the standard TCP. This process will be in effect until the packet loss stops. Of course, halving the *cwnd* will definitely reduce the network throughput which has to be compromised in order to prevent further packet drop. Once the packet loss stops, the slow-start algorithm kicks in, beginning with exponential growth. When the window sizes are small, the recovery time is quite fast. But, at higher transfer speeds the recovery time becomes very high.

The original idea of TCP-Scalable is to scale the performance of standard TCP at higher and lower transfer speeds. Further, the performance of TCP-Scalable is bound to vary due to differences in network characteristics of different networks, such as wired network, wireless network, MANET and VANET. Also, if one tries to use TCP on

VANET, then the possibility of TCP interfering with the existing safety messaging arises. In this section, a simple idea is presented to resolve smoothly the competing tendency of TCP-Scalable against safety applications.

The TCP-Scalable algorithm handles congestion control in a way different from standard TCP. It uses MIMD algorithm to calculate *cwnd* growth. Because of MIMD policy, its *cwnd* growth is very aggressive in both increase and decrease directions. For a network that has long delays (e.g. satellite), this protocol cannot produce stability. Each loss of packet in TCP-Scalable causes *cwnd* to reduce to a fraction of 1/8 rather than 1/2, as used in the standard TCP, until packet drop stops. The moment packet drop stops, the window is increased with fixed rate; one packet is added after the receipt of every 100 successful ACKs. In this way, the recovery time in high bandwidth link reduces to a great extent. However, in the standard TCP the packet addition rate is based on the inverse of current *cwnd* size, and in such a situation large window takes a long time to recover.

The main objective of TCP-Scalable [75] is to improve the loss recovery time of the standard TCP. It quickly recovers the window size after the congestion sets in. TCP-Scalable achieves this objective by resetting the *cwnd* to a higher value than the standard TCP. Formulation of TCP-Scalable is basically derived from and motivated by the principle of TCP-HighSpeed. In the standard and HighSpeed TCP, the recovery time for each loss of packet is relative to RTT and *cwnd* size. However, in TCP-Scalable the recovery time of packet loss is relative to RTT only. The slow-start phase of TCP-Scalable is not modified and it is in accordance with standard TCP, where the *cwnd* growth is increased by one packet (*cwnd*=1) for each received ACK. The *cwnd* continues to grow until either window size exceeds its initial *ssthresh* level or packet loss occurs. The main modification of TCP-Scalable comes from congestion avoidance phase as given below:

For each successful received ACK in a RTT, where congestion has not been detected, the *cwnd* growth is made as:

Traditional TCP	TCP-Scalable
cwnd = *cwnd* + 1/*cwnd*	*cwnd* = *cwnd* + 0.01

In the event of congestion in a RTT, the *cwnd* multiplicative factor is reduced as:

Traditional TCP	TCP-Scalable
cwnd = *cwnd* – 0.5 * *cwnd*	*cwnd* = *cwnd* – 0.125 * *cwnd*

NS2 TCP Linux: It is a Linux TCP implementation for NS2, which contains a collection of different TCP congestion control algorithms. In NS2 implementation, the TCP congestion control algorithms loosely follow the real Linux TCP implementation of the same protocols, but it can produce results comparable to real Linux TCP suite. However, NS2 architecture tightly follows ISO-OSI model. Implementation of a new protocol and its simulation under NS2 essentially follow these four important steps:

- Implementing the new protocol by coding algorithm in C++
- Configuring and defining simulation parameters in OTcl scripts
- Running the simulation to generate discrete events which are stored in trace files
- Analysing the generated trace files using some metric

In order to implement a new protocol in NS2, the simulator usually requires C++ coding for the operation of protocol; after that, it needs to update NS2 OTcl configuration files, which enables NS2 to identify the newly added protocol and its parameters. Further to note is the fact that the C++ code also describes which parameters and procedures are to be made available for OTcl scripts. The implementation of a new TCP protocol under NS2 TCP Linux includes the following steps.

- The header files to link the implementation to TCP-Linux
- Definition and declaration of parameters, if any, needed
- Implementation of at least two congestion control functions defined in **struct tcp_congestion_ops:**
  - cong_avoid
  - ssthresh and
  - min_cwnd

- A static record of **struct tcp_congestion_ops** to store the function calls and algorithm's name.
- A module initialization function

Table 5.1 shows four important variables that control the behaviour of TCP-Scalable algorithm.

Variable Name	Meanings	Type
snd_ssthresh	Slow-Start threshold. Phase is in slow start if snd_cwnd is less than this.	Unsigned (32 bit)
snd_cwnd_cnt	Fraction of congestion window. It is a counter which is used to slow down the rate of increase, once slow start threshold is exceed.	Unsigned (16 bit)
snd_cwnd	The size of the congestion window.	Unsigned (32 bit)
snd_cwnd_clamp	Upper bound of the congestion window. This is the maximum size that snd_cwnd can grow.	Unsigned (16 bit)

Table 5.1: Variable that control behaviour of TCP-Scalable

The following are the two important functions that define the behaviour of TCP-Scalable algorithm:

(1) void (*cong_avoid) (struct tcp_sock *sk, unsigned int ack, unsigned int rtt, unsigned int in_flight, int good_ack);

This function is called every time an ACK is received and the congestion window is then increased.

- ack: is the number of bytes that are acknowledged in the latest ACK
- rtt: is the RTT measured by the latest ACK
- in_flight: is the packet in flight before the latest ACK
- good_ack: is an indicator whether the current situation is normal

(2) unsigned int (*ssthresh) (struct tcp_sock *sk);

This function is called when the TCP flow detects a loss. It returns the slow-start threshold of a flow after a packet loss. The pseudo code for original TCP-Scalable's *congestion avoidance* and *ssthresh* function body is shown below.

Pseudocode: Original TCP-Scalable

```
//Tom Kelly’s TCP-Scalable
cong_avoid (...)
if snd_cwnd <= snd_ssthresh then
   start_slow_start ();
else snd_cwnd_cnt++;
     tmp = min(snd_cwnd, AI_CNT)
  if snd_cwnd_cnt > tmp then
      if snd_cwnd < snd_cwnd_clamp then
         snd_cwnd++;
         snd_cwnd_cnt = 0;
      end if
  end if
end if
end of function
ssthresh (...)
 tmp = snd_cwnd - (snd_cwnd >> MD_SCALE)
if snd_ssthresh > min then
   snd_ssthresh = tmp
   return snd_ssthresh
else
   snd_ssthresh = min
   return snd_ssthresh
end if
end of function
```

Note: The values of AI_CNT=50 and MD_SCALE=3 were derived from the recommended values of TCP-Scalable. In NS2 implementation for AI_CNT, 50 is used instead of 100 to account for delayed ACK. Which means TCP-Scalable in this implementation will increase its congestion window by 1 for every 50 received ACK.

5.4 Design & Implementation of Optimized vScalable-TCP Algorithm

During the execution of slow-start phase, the cause of unacknowledged segment might come from network congestion. This assumption is true in case of guided media network and static wireless network; however, the loss of a segment in MANET or VANET could be because of poor PHY and MAC layer transmission quality. The slow-start mechanism under performs in wireless network where signal reception quality is not good and reliability is not certain. The performance of slow-start imposed by TCP congestion control mechanism in VANET is poorer because of short-lived connections and frequent disconnections. If the session is for web browsing, then the web browser would generate several successive short-lived sessions to a web server as well as open and terminate the session for each file or page requested. These phenomena mostly occur in slow-start phase of congestion control, resulting in poor response time.

The so-called ‘slow-start’ of *cwnd* growth is, in reality, very aggressive in the standard TCP and is even more aggressive than the congestion avoidance phase. Before the slow-start algorithm was introduced in the TCP protocol, the initial pre-congestion avoidance phase was even faster. This is the reason it was called slow-start. In the slow-start phase of standard TCP, the growth of *cwnd* is increased by 1 MSS for each received ACK. This process doubles the growth of *cwnd* for each RTT leading to a sharp increment of window size in the network that has large BDP. In one respect, this is a favourable procedure because it rapidly occupies the available bandwidth. However, if it is not checked or stopped before the receiver buffer overflows, then depending upon the window state, 10 to 100 packets might be lost in a single RTT. This huge packet drop could precipitate severe congestion, leading to high degree of performance degradation of TCP as well as heterogeneous time-sensitive flows which traverse in parallel. Also, if the packet dropping is the result of network congestion, then pumping more data (that starts from the slow-start phase) will only make matters worse with increased network congestion which is obviously not desirable for any type of use case.

During the congestion avoidance phase *cwnd* reaches a stable state where TCP sends data at the maximum transmission rate and bandwidth is not greedily occupied.

Standard TCP with linear *cwnd* growth adds 1 packet after the receipt of each ACK in the congestion avoidance phase. As discussed earlier, if during the congestion avoidance phase the packet loss is due to triple duplicate ACKs then fast retransmit and fast recovery mechanisms are triggered where *cwnd* does not shrink to 1 segment, and as a result performance of TCP based application does not reduce much. So, the impact of controlling of the packet pumping anywhere at the stage of congestion avoidance phase will severely degrade the performance of TCP because fast-recovery and fast-retransmit algorithms are already fully at work.

Bearing in mind the huge packet drop in a single RTT, and the limitation of congestion avoidance phase, and the performance of BSM degradation in the presence of the TCP flow which is amply attested by previous comprehensive performance analysis made in this work, it was thought prudent to tweak the slow-start phase of TCP-Scalable.

If the slow-start mechanism is not controlled in the event of huge packet loss, then the bandwidth will be unnecessarily occupied and wasted by the packet which is anyway likely to be dropped somewhere sometime during its propagation. Also, frequent calls to slow-start only promotes aggressive occupation of bandwidth and as a result the flow gets throttled and the safety application suffers.

To bring about a resolution of this impasse, it appears imperative that the behaviour of slow-start should be controlled and tweaked cleverly in some way and in the proposed vScalable-TCP algorithm - the details of which follow - an innovative technique is devised to do just the same i.e. control or adjust the behaviour of slow-start in a suitable way, while at the same time maintaining the structure and integrity of the normal TCP-Scalable algorithm.

In practice, slow-start is more aggressive in VANET environment and has negative effect on the performance. So, frequent calls to slow-start phase will actually trigger congestion more under VANET scenario. This phenomenon is observed during the experiments (in section 4.2) with original TCP-Scalable algorithm. Even though the average throughput seems to be increasing due to the slow-start triggered by *snd_ssthresh* in case of normal TCP-Scalable, actually the total dropped packets is also getting increased, which means that some of the valuable bandwidth of VANET is sacrificed for the sake of an incremental benefit in throughput.

According to the proposed model, if the packet drop due to congestion is high, then the function *ssthresh* will be called very frequently, so that the *DropInterval* will asymptotically approach zero and eventually gets truncated to zero. At this point, TCP slow-start is temporarily suspended until next restart to avoid aggressive slow-starts at the very congested situation. If the drop interval is above zero and less than or equal to one, then the *snd_ssthresh* is reduced much, otherwise it is reduced less. These maximum and minimum values can be selected based on the use of TCP flows in a practical scenario.

Any TCP algorithm will increase the sender side window size if it successfully completes sending a packet (the *cong_avoid* function is called during this time). The TCP algorithm will call the *ssthresh* function in the event of any packet loss. In that function, generally the TCP send window size will be reduced. How much it will be reduced depends upon the type of TCP algorithm used and this constitutes the unique characteristic of each TCP algorithm. But, if that reduction of TCP send window size reaches a particular minimum value (threshold), then the TCP slow-start will get triggered. In the modified algorithm, the slow-start triggering is avoided by setting a negative *snd_ssthresh* to stop any further drop of packet - in other words, the TCP flow gets stopped if there is too much loss. Later, the application should restart the connection (re-transmission) if it needs to resume.

The following Algorithm-1 shows modified *ssthresh* function which will replace the one in the original TCP-Scalable when deployed in VANET environment. Since the modification is made in the context of VANET architecture, the prefix **‘v’** is given to TCP-Scalable and renamed it as vScalable-TCP.

In the modified vScalable-TCP algorithm, all the parameters were chosen based on the observation of the behaviour of normal TCP-Scalable under proposed VANET scenario.

Algorithm-1: Modified vScalable-TCP

```
 PreviousTime=0
 DropInterval=0
 MAXSHIFT=4
 MINSHIFT=2

cong_avoid(...) //same as original TCP-Scalable

ssthresh(...)
 CurrentTime = tcp_time_stamp/1000.0;
 DropInterval = CurrentTime - PreviousTime;
if DropInterval <= 0 then
    //suspend slow-start until next restart
      snd_ssthresh = -1;
else if DropInterval <=1 then
    //reduce much
      snd_ssthresh = tp -> snd_cwnd >> MAXSHIFT;
else
    //reduce little
      snd_ssthresh = tp -> snd_cwnd >> MINSHIFT;
 PreviousTime = CurrentTime;
 return snd_ssthresh
end of function
```

5.5 The Performance of Non-safety Application over vScalable-TCP

In this section, the performance results of the vScalable-TCP for non-safety TCP based file transfer application are presented. Similar to previous study, there is a main safety short messaging application periodically broadcasting in the background during the TCP based file transfer.

Scenario-2 with its associated parameters and the four metrics (throughput, PDF, E2ED and packet drop) as defined and discussed in Chapter 3 are taken up for the performance study of TCP-Scalable and vScalable-TCP. The simulation result shows the performance of TCP file transfer application in a freeway scenario. Though the main goal is to compare the performance of TCP-Scalable and the proposed vScalable-TCP, some other TCP algorithms are also included to get a general picture about improvement in the proposed algorithm. A few vehicles in the scenario are accessing FTP resources from RSU in multi-hop fashion, while at the same time all the 50 vehicles are periodically broadcasting safety services at varying intervals. Since the data traffic is comprised of both safety messages and FTP over TCP, the BSM broadcast data traffic is assumed to occupy the bandwidth based on message interval rate, whereas the bandwidth for TCP traffic is based on congestion window dynamics.

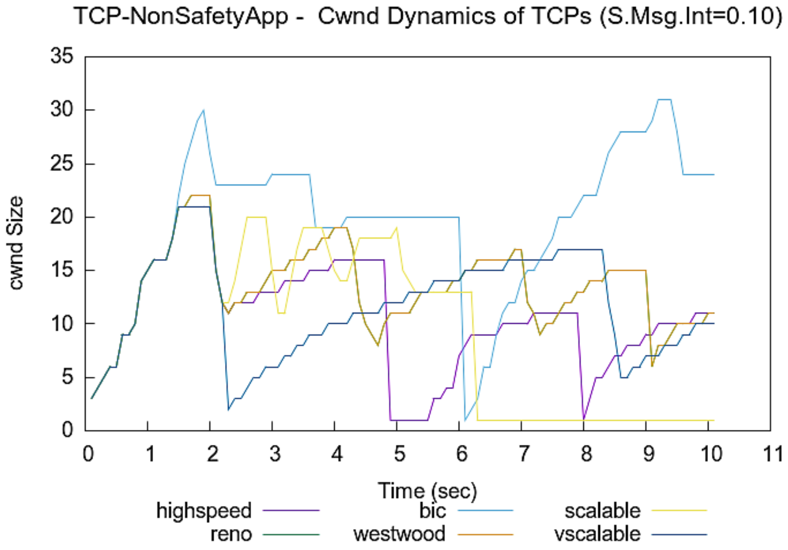


Figure 5.1: cwnd dynamics of TCP variants

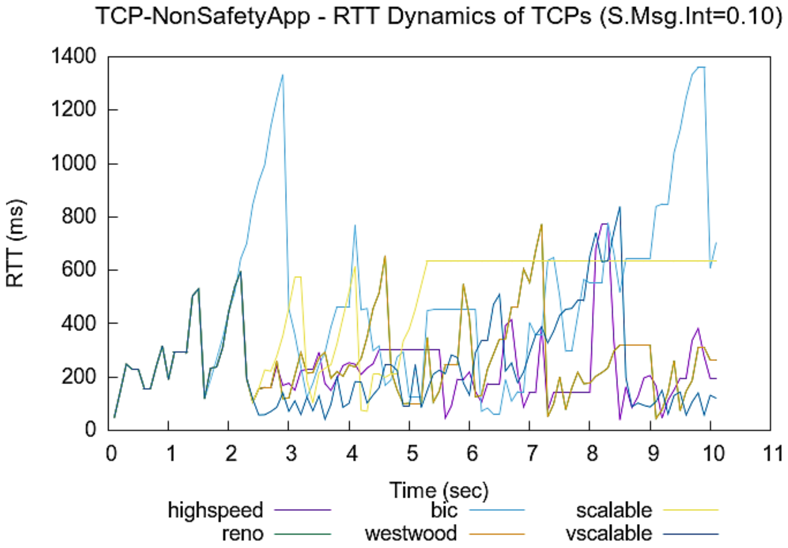


Figure 5.2: RTT dynamics of TCP variants

The line chart in the Figure 5.1 shows congestion window dynamics *vs* elapsed time of different TCP algorithms while running a non-safety file transfer application for a short interval of time. There is one main safety messaging application in the background, periodically broadcasting safety messages at the interval of 0.10s. Understanding the *cwnd* dynamics, which is not the average of multiple TCP flows, is not reliable and straightforward. When a connection is established, the *cwnd* is set to the MSS allowed on that connection and after that slow-start begins. The variation in the *cwnd* of vScalable-TCP is governed by MIMD policy, which means that if all the segments are successfully sent and the corresponding ACKs are successfully received back at the sender side on time, then some constant term is added to the window size. The window keeps on growing linearly until a RTO or duplicate ACK occurs or the receiver reaches buffer overflow in the congestion avoidance phase.

As shown in the graph, the performance of the proposed vScalable-TCP is somewhat better than TCP-Scalable in terms of stability. Till the first 1.5s of connection time, the exponential growth is evident, which comes from slow-start phase of modified vScalable-TCP. The dynamics of TCP-Scalable algorithm clearly reflects the modification, where the first slow-start (at 1s) followed standard TCP which exhibited aggressive exponential increment. But at 2s, because of triple duplicate ACKs or RTO expiry, there is multiple decrease of the *cwnd*. And at around 2.2s, the suppression of slow-start and linear *cwnd* growth become evident, which reflect the effect of the optimized version vScalable-TCP. This indicates that vScalable-TCP efficiently utilizes the *ssthresh* to help recover more smoothly in the event of packet loss compared to all the other TCP variants.

The *cwnd* continues to increase starting from 2.2s as the ACKs are being continuously received until around value of 8.1s. The stable *cwnd* growth indicates smooth transfer of data with minimal packet drop. Though the vScalable-TCP *cwnd* growth is slow, it is stable for a longer duration of time in comparison with original TCP-Scalable and all the other algorithms, which is precisely what is required for VANET environment. Thus, when compared with all the simulated TCP variants, the performance in terms of stability of the *cwnd* growth is undoubtedly superior in this new approach.

The line chart in the Figure 5.2 shows the RTT dynamics of different TCP algorithms, while running non-safety TCP file transfer application for a short interval of time. RTT dynamics presented is also not an average of multiple TCP transfers but the result of only one TCP flow. The variation in RTT is in correspondence with the variation of the *cwnd* growth, where the sender side estimates the current bandwidth consumed against each RTT. The performance of vScalable-TCP is somewhat better in terms of RTT when compared with original TCP-Scalable. An almost uniform lowest RTT dynamics in vScalable-TCP, starting from 2.1s till 8th s, is a clear indication of fair utilization of bandwidth. The high RTT at 2s and 8s is the result of triple duplicate ACKs or RTO.

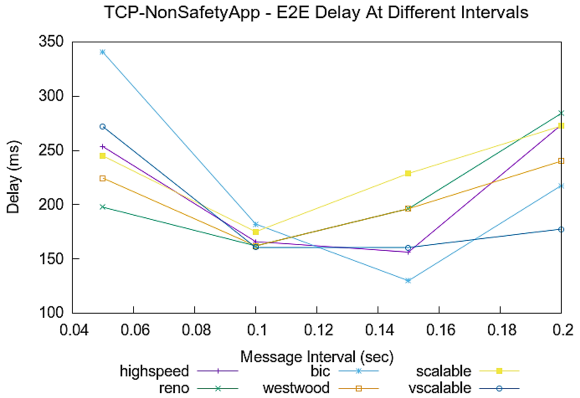


Figure 5.3: E2ED of TCP variants *vs* BSM intervals

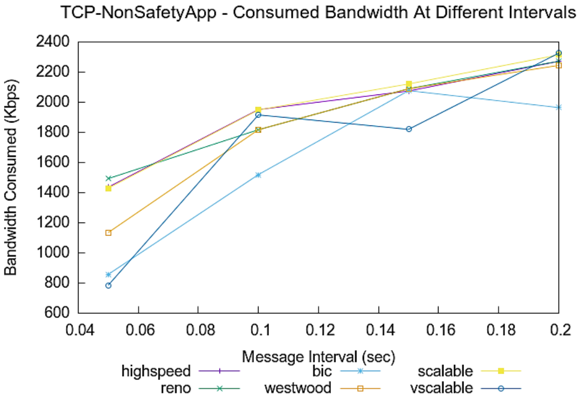


Figure 5.4: Average Throughput of TCP variants *vs* BSM intervals

The Figure 5.3 shows the performance in terms of average E2ED achieved by the TCP non-safety file transfer application with respect to background safety message intervals rate. The E2ED measures the average time taken for a data packet transmitted from the source vehicle to the destination vehicles or RSU. It is generally understood and accepted that low E2ED associated with optimal throughput is most favourable, as it provides robustness to any system that includes TCP algorithm to handle congestion events. The E2ED line chart of vScalable-TCP shows an interesting linear trend starting from 0.1s till 0.2s of message interval rate in comparison with TCP-Scalable and the other TCP variants.

Due to aggressive behaviour, the TCP-BIC curve shows that significant performance decrement occurs at 0.15s while it is acceptable for vScalable-TCP and TCP-HighSpeed. However, very similar dynamics prevails for almost all the simulated algorithms at the 0.1s of message interval rate. The overall E2ED curve reveals that when the message interval lengths are too small or too large the results are not good, as the E2ED gets higher for all the algorithms. Further, the result reveals that medium BSM interval rate (0.10s till 0.15s) can be considered suitable for BWE for making better and fair utilization of available bandwidth.

Compared with all simulated TCP variants, the performance of modified vScalable-TCP is better in terms of E2ED metric, being almost uniform and having average low latency, right from 0.1s till 0.2s of message interval rate. The average 175ms of latency provided by vScalable-TCP is just sufficient to support basic infotainment applications that can complete data transmission within 10s of connection to RSU. On the whole, vScalable-TCP dynamics shows good trade-off between prudent utilization of bandwidth and suitable E2ED.

The performance in terms of average consumed bandwidth of TCP variants with respect to BSM broadcast intervals rate is displayed in Figure 5.4. It is evident from the line chart that with the increase of BSM broadcast interval rate, TCP average throughput is also increasing. The reason for throughput increase or decrease is BSM data traffic, which is coexisting with the non-safety application over the same channel. The broadcasting delay is achieved through message interval rate. The broadcasting delay is lowest and highest at 0.05s and 0.2s respectively, where low broadcasting delay signifies higher number of broadcasts.

The delay in message broadcasting frees the bandwidth for the defined interval of time, which is being fully utilized by the TCP application; as a result, low broadcast interval rate becomes inversely proportional to TCP throughput. However, during the broadcast time, TCP still occupies the amount of bandwidth, based on the type of congestion control mechanism applied. Apart from broadcast interval frequency, the rate of throughput can be associated with E2ED. With the decrease of E2ED, it is obvious that the higher the bandwidth utilization, the better the throughput.

The average throughput of vScalable-TCP has decreased almost 50% in comparison with TCP-Scalable during 0.05s of message interval rate, but at 0.10s, both TCP-Scalable and vScalable-TCP maintain the same level of throughput. In case of 0.05s of message interval rate, the bandwidth becomes congested, as it is mostly occupied by safety messages along with some TCP data traffic; as a consequence, there could be a high number of TCP packets dropped.

Based on the optimized vScalable-TCP algorithm, if a high number of packets are dropped, then slow-start is temporarily suspended until next start. And, if there is less packet loss then the *cwnd* is reduced as per the defined value in the algorithm, in order to prevent further loss and ensure optimum flow. So, the reduced throughput of vScalable-TCP at 0.05s comes from the call to *ssthtresh* function in the event of packet loss – when the *cwnd*, which is directly associated with throughput, is reduced - as well as from E2ED at 0.05s message interval which is higher than original TCP-Scalable. Also, vScalable-TCP reduces its throughput a little to avoid congestion, thereby releasing some of the network resources. Hence the increase in E2ED.

In the case of 0.10s message interval rate, the bandwidth is fairly utilized where the E2ED is less than TCP-Scalable. On the whole, if the broadcast interval rate of 0.05s is needed for a particular safety application that is coexisting with some other important non-safety application like electronic toll tax collection or local service information, then the achieved average of 800 kb/s data transmission rate is adequate.

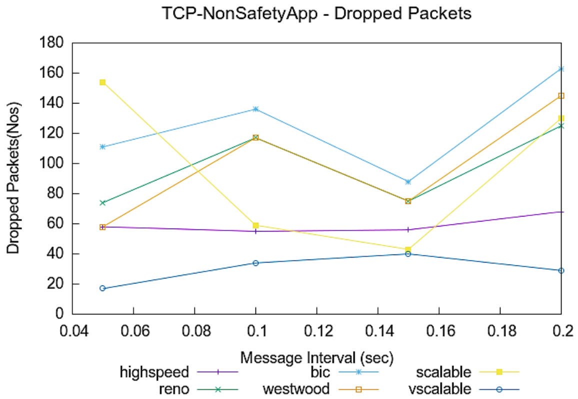


Figure 5.5: Total Dropped Packets of TCP variants *vs* BSM intervals

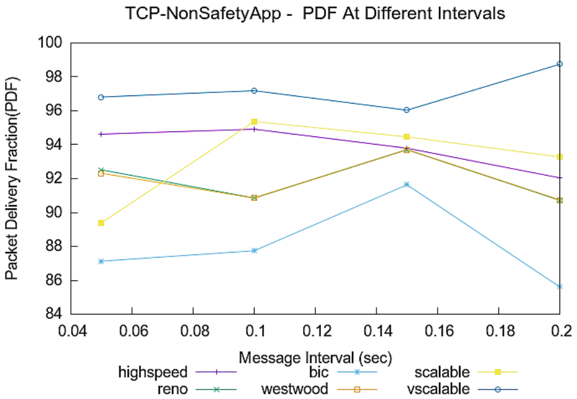


Figure 5.6: PDF of TCP variants *vs* BSM intervals

The performance results in terms of dropped packets and PDF of non-safety (TCP) file transfer application with respect to varying background safety message intervals rate are depicted in Figure 5.5 and Figure 5.6 respectively. The data packets drop metric is complementary to PDF, where the number of dropped packets decreases when the PDF increases and vice versa. The reason for packet drop could be wireless channel error, frequent disconnection, or buffer overflows at the intermediate or destination vehicle.

If a packet loss occurs due to receiver buffer overflow or propagation delay, the vScalable-TCP tunes itself to adapt the congestion window based on the network status. The vScalable-TCP dropped fewer packets because of the nature of the proposed algorithm. In the event of network congestion, it simply minimizes packet dropping by regulating the TCP flows that require activating the new *ssthreshold* function. Comparing PDF and packet dropped results, the duality in the characteristics of vScalable-TCP and TCP-Scalable becomes evident. As shown in the graph, vScalable-TCP outperformed TCP-Scalable and all other compared algorithms in terms of both packets dropped and PDF.

5.6 A Study on the Impact of vScalable-TCP on Periodic BSM

In the previous section 5.5, the performance of vScalable-TCP using various performance analysis metrics has been evaluated. It is found that the performance improvements in terms of E2ED, packet dropped rate and PDR are achieved by sacrificing average consumed bandwidth, and specifically at the message interval rate of 0.05s and 0.15s. Since DSRC is primarily concerned with V2V safety messaging applications, it is necessary to evaluate the impact of modified vScalable-TCP on BSM application in the presence of infotainment application so that its real characteristics are understood and its suitability for VANET assessed. To proceed in this direction, this section presents the impact of vScalable-TCP on safety message application at different message broadcast interval rates.

Similar to previous study, a main safety short messaging application that is periodically broadcasting is kept as high priority traffic in the network, with a TCP based file transfer application running in the background as a low priority data flow. Scenario-2 with its associated parameters and the four performance analysis metrics (throughput, PDF, and E2ED) as described in Chapter 3 provides the backdrop for this study also.

The simulation results show the performance of BSM application with respect to TCP file transfer application in a freeway scenario. Though the scope is to compare the impact of TCP-Scalable and the proposed vScalable-TCP, some other TCP algorithms are also included to get an overall idea about comparative performance improvement in the proposed algorithm. In the given scenario, all the 50 vehicles are periodically broadcasting safety services at varying intervals, while at the same time a few vehicles are accessing FTP resources in multi-hop fashion from an infostation. Since the data traffic is comprised of both broadcast BSM and FTP over TCP, again, the BSM broadcast data traffic is assumed to occupy the bandwidth based on broadcast interval rate, whereas TCP traffic is based on choice of TCP’s congestion window dynamics.

The line charts in this section actually exhibit the performance of BSM application with respect to TCP file transfer application. These graphs show how much TCP flows affect the performance of safety messaging application under defined sparse vehicular freeway scenario.

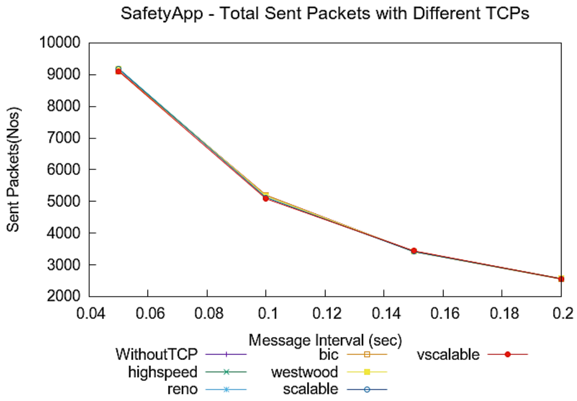


Figure 5.7: BSM Sent Packets in the presence of TCP variants

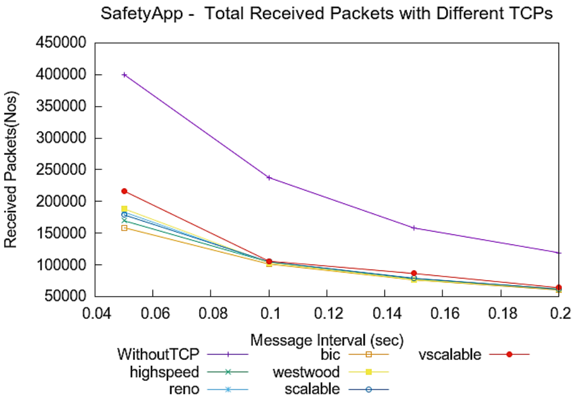


Figure 5.8: BSM Received Packets in the presence of TCP variants

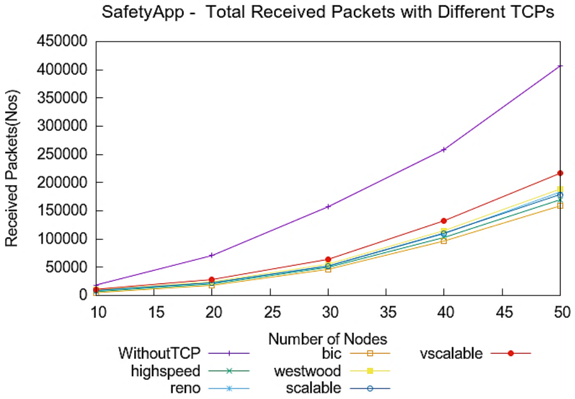


Figure 5.9: BSM Received Packets in the presence of TCP variants & No. of nodes

Figure 5.7 shows the performance in terms of total sent packets achieved by the BSM at different broadcast intervals rate and with different TCP flows. As seen in the graph, the send performance of the messaging application is not at all affected by any kind of TCP flows, but as the broadcast interval rate increases the number of BSM packets transmitted in the presence of each TCP algorithm decreases. The reason for this phenomenon is obvious viz. broadcast interval rate is inversely proportional to message sending rate. On the contrary, in Figure 5.8 the performance in terms of total received packets count is affected by all kinds of TCP flows, and the impact is considerably high at higher broadcast interval rate. But, in the case when there is no TCP flow (WithoutTCP curve), the whole bandwidth is occupied by only BSM traffic; thus performance of BSM is independent of TCP, as expected.

In the plot showing the average received packets, because of less aggressive but stable *cwnd* growth of vScalable-TCP, one observes the safety messaging application receiving many more packets than TCP-Scalable and other TCP variants. The lesser the impact, the better and fairer the bandwidth utilization by BSM in the presence of vScalable-TCP flow, as is evident at 0.05s till 0.08s of message interval rate.

The performance details in terms of total received packet with respect to increasing number of vehicles is shown in Figure 5.9. The received packet count is generated using the BPSK modulation and broadcasting of safety messages at the interval of 0.05s. As can be seen, the received packet count is directly proportional to the number of mobile nodes. Unlike Figure 5.8, the received packet count in Figure 5.9 clearly indicates that the performance of BSM in the presence of vScalable-TCP is unaffected by the growing number of vehicles and this is certainly encouraging. .

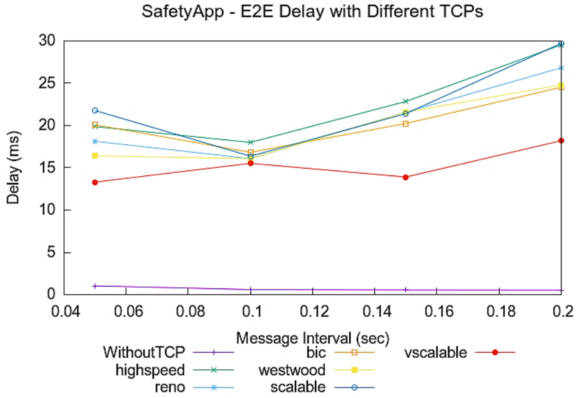


Figure 5.10: E2ED of BSM in the presence of TCP variants

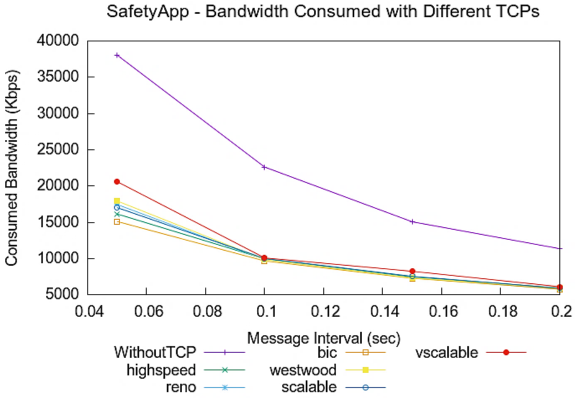


Figure 5.11: Average Throughput of BSM in the presence of TCP variants

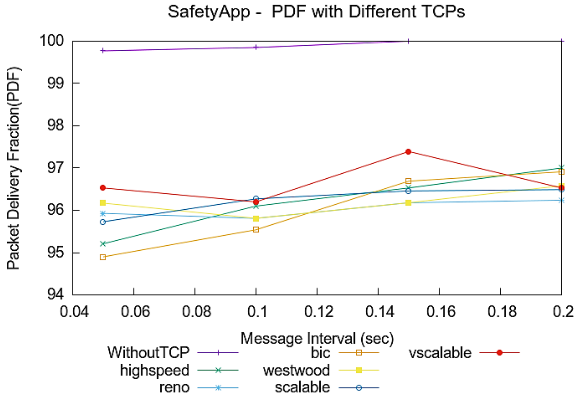


Figure 5.12: PDF of BSM in the presence of TCP variants

The performance details in terms of E2ED achieved by the safety messaging application at different broadcast interval rates and with different TCP flows are presented in the Figure 5.10. As shown in this graph for E2ED in the presence of vScalable-TCP and other TCP variants, the BSM flow is less affected and within the required average delay of 0.1s, at each message interval rate. In comparison to TCP-Scalable, the vScalable-TCP performance at 0.05s and 0.15s is distinct with less impact on BSM flow which is suitable for general safety use cases. But in the presence of high speed vehicles, where time critical safety message broadcasting at the interval of 0.05s is desired, vScalable-TCP, whose E2ED is less than 15ms, will definitely outperform in comparison to TCP-Scalable and other TCP variants. On the whole, because of low latency, the bandwidth utilization by BSM in the presence of optimized vScalable-TCP is better than TCP-Scalable and other algorithms.

Shown in Figure 5.11 is the performance in terms of bandwidth consumed by BSM application at different safety message broadcast intervals rate with respect to TCP variants. The correlation with received packets and consumed bandwidth is evident from Figure 5.8 and Figure 5.11, where high number of received packets resulting in better throughput is obvious. The TCP-Scalable dynamics shows aggressive behaviour in the graph, so it occupied bigger portion of bandwidth, which results in lower bandwidth utilization from BSM traffic, and as a result end-to-end latency becomes high. The modified vScalable-TCP exhibits less aggressive behaviour that positively helps to achieve higher and faster bandwidth utilization from BSM traffic. As a consequence, one obtains low latency and better throughput and it is naturally along expected lines.

The line chart in Figure 5.12 shows the performance in terms of PDF achieved by the safety messaging application at different safety message intervals and with different TCP flows. The PDF graph is not much affected by any kind of TCP flow, as there is low variance at each message interval rate and more than 95% of delivery ratio is evident in every TCP algorithm. But, in the presence of vScalable-TCP, overall PDF performance of the safety messaging application with respect to both variance and ratio is much better than all the simulated algorithms.

5.7 Chapter Summary

It is no surprising the performance of TCP file transfer under VANET is very much affected by the safety message broadcast interval rate. And, similarly, the DSRC safety short messaging application is also affected by the TCP flows. The impact on each other is due to BWE that comes from broadcast interval rate and congestion control mechanism used by BSM and TCP respectively. Because each TCP algorithm adopts different approach in tackling congestion events, the magnitude of impact on core safety messaging services accordingly varies. As discussed earlier, beacon based BSM broadcast interval rate incurs wastage of bandwidth. If one tries to capitalize on the unutilized bandwidth of a channel for a non-safety infotainment purposes using TCP, then it potentially amounts to *stirring up a hornets' nest,* and therefore it certainly has the potential to adversely affect the high priority safety messaging traffic. So, if there are requirements to use TCP for file transfer or delay-insensitive applications on VANET, when core BSM is already running, then one has to somehow nudge low priority TCP traffic to behave in a non-intrusive manner.

The main objective of DSRC is, undoubtedly, the deployment of safety critical application to prevent vehicle related accidents on road. So, while using non-safety services, one never afford to compromise on the efficiency and effectiveness of safety applications, if the former are required to coexist with the later over the same DSRC protocol.

In this work, a different tack is taken and an innovative solution is offered to handle the slow-start issue at congested situation. In this model, at a highly congested state, the slow-start is suppressed temporarily to avoid further congestion and further packet drops. The main motive for effecting this modification on TCP is to regulate its performance so that it will not hurt unduly the existing safety message flows. This idea has been successfully implemented on TCP-Scalable and a new algorithm named vScalable-TCP is derived.

The proposed, improved and optimized vScalable-TCP algorithm performs better without affecting much the integral periodic safety messaging applications. Moreover, rather surprisingly, it has significantly improved the performance of the main safety application in terms of PDF, E2ED and throughput. However, it should be noted that in order to maintain priority to DSRC time-sensitive application the overall average throughput of vScalable-TCP has to be brought down from its original performance. And this trade-off is indeed worthwhile.

Based on the extensive simulation carried out earlier and the results obtained there from, it is clear that with the 3 Mbps link bandwidth it is difficult to further enhance BSM throughput, but one can extend the (re)search to look out for the possibility with another most suitable congestion control algorithm. In this way, the adaptability and robustness of the proposed algorithm in other TCP variants can also be evaluated. So, in the next stage of this study, the same optimization strategy is tried with a little modification in another congestion control algorithm *viz.* TCP-LP to assess its suitability and efficacy. And this is taken up in the next chapter.

Chapter 6

vLP-TCP: An Optimized Low Priority TCP Algorithm to Share Bandwidth Cooperatively with Safety Application

6.1 Introduction

Previous Chapter 5 described vScalable-TCP, a new formulation of TCP-Scalable, optimized by tweaking *ssthresh* function in the original version of TCP-Scalable. In comparison to TCP-Scalable and other loss-based TCP congestion control algorithms used in this work, the performance improvement of vScalable-TCP in terms of PDF, packet loss rate, E2ED and throughput of safety messaging flow is found to be encouraging. But the average throughput of TCP flow is compromised at the message interval rates of 0.05s and 0.15s. The freeway sparse vehicular scenario under consideration consists of only three nodes that have TCP connections to RSUs in multi-hop fashion. The reduced average throughput obtained at the message interval of 0.05s is around 700 kb/s only, which is just sufficient to fulfil the basic infotainment application requirements such as electronic toll collection, small file download, roadside service information and music download. Essential vehicular service type of applications being usually short in size, the bandwidth consumed will not be much on an average, even if a large number of vehicles try to concurrently access such type of use cases. In general, DSRC based nomadic vehicular stations in a freeway scenario do not expect high number of TCP connections. But, if the link bandwidth is limited to 3

Mbps and a large number of vehicles try to access bandwidth hungry-applications, then it is obvious the throughput will suffer further.

In this chapter, the TCP-LP algorithm is examined critically and an optimized vLP-TCP is designed that will also be suitable for deployment over VANET environment. The proposed vLP-TCP is expected to give better performance without sacrificing much the existing safety message flows. The same modelling approach that was applied in vScalable-TCP which tweaked the *ssthresh* function in the previous chapter, is adopted here too.

The main objective of employing the same approach in TCP-LP is to evaluate the effectiveness and adaptability of the same modelling technique that is so successful in vScalable-TCP and to see if the same framework can fit in with any congestion control mechanism and make it suitable for VANET. Though, in general, the same optimization mechanism is unlikely to be uniformly applicable to all existing TCP variants, because each congestion control algorithm has its own unique control on *cwnd* growth, still tweaking slow-start in the event of huge packet loss can be tried and experimented with in most of the TCP variants.

The TCP-LP is the only algorithm in the TCP congestion control family that has the mechanism to treat the background TCP traffic in a transparent and non-intrusive manner with the other TCP flows. So, it is obvious from the nature of TCP-LP that the performance of TCP flow should be better in comparison to other TCP variants. But in the present scenario, in the presence of TCP-LP flow, the other low priority TCP traffic does not exist; instead broadcast based small payload of data is traversing the network, so some kind of anomalous behaviour is expected. The degree of anomaly depends upon various parameters including data traffic types and pattern of propagation, number of simultaneous users, link bandwidth, link quality, payload size, network environment and of course RSS.

6.2 Analysis & Implementation of TCP-LP on NS2 Linux TCP

TCP-Low Priority (TCP-LP) [78] is a sender side distributed congestion control algorithm. The sole objective of TCP-LP is to utilize only the unused excess network bandwidth in a non-intrusive manner in contrast with fair bandwidth share policy used by TCP-Reno. Two unique congestion control features adopted by TCP-LP are the implementation of one-way packet delay which detects congestion at an early stage rather than after packet loss occurs, as used by TCP-Reno, with the other feature being the provision of transparent congestion avoidance.

TCP-LP has the capability to maintain transparency with standard TCP flow, even if the former takes long RTT than the latter. To achieve low priority service in the presence of TCP traffic, it is necessary for TCP-LP to infer congestion earlier than plain TCP. In the standard TCP, packet drop is considered as a harbinger of congestion, but TCP-LP uses one-way packet delays as an indicator to infer early the impending congestion. It measures one-way packet delays and employs a simple delay threshold-based method for early inference of congestion. The net result of this early congestion inference estimation is the detection of congestion only when data move from source to destination, and at the same time, this avoids committing *false positives* i.e. forbids false early congestion indication from backhaul background or cross traffic. In this way, TCP-LP and standard TCP implicitly coordinate in a distributed manner to provide expected priority levels.

TCP-LP uses the TCP time-stamp option [164] in order to gather samples of one-way packet delay. Each TCP packet carries 2 time-stamp fields of four bytes each. In order to measure one-way packet delay, a source node that has TCP-LP agent, time-stamps one of these fields with the current clock value when the data packet is transmitted in the network, while destination node repeats back sender node time-stamps value and also its own current time in the ACK packet.

In addition to one-way packet delay method, TCP-LP also has implemented a novel congestion avoidance mechanism that changes AIMD algorithm in standard TCP by adding inference phase, and has adopted modified back-off policy in order to fulfil three objectives. The prime objective is quick reduction of *cwnd* with the help of back-off procedure in the event of congestion caused by non-TCP-LP flows. The second one is quick utilization of unused network excess bandwidth unoccupied by non-TCP-LP, and finally the third crucial objective is to provide fairness among the multiple TCP-LP flows.

Further, TCP-LP algorithm has adopted a procedure to provide faster response time for interactive applications in low speed access link, and at the same time to smoothly allow background traffic such as file transfer to proceed unhindered. In other words, what this means is that TCP-LP permits low priority services to use all available excess bandwidth, while remaining transparent to other TCP flows.

According to theory, TCP-LP efficiently utilizes the network excess bandwidth either in single or multiple flows. When using multiple flows, the TCP-LP fairly shares the bandwidth among them, at the same time making available excess bandwidth to low priority traffic. In case of bandwidth-hungry application flows, TCP-LP is able to efficiently utilize substantial amounts of unused bandwidth. These features of TCP-LP make it an ideal candidate to experiment with BSM flows in a VANET environment.

NS2 TCP-LP on Linux:

In Chapter 5 the universal procedure to implement any new protocol in NS2 TCP Linux is described in detail (*vide* 5.3). The same procedure is followed here for the implementation of TCP-LP algorithm. In the TCP congestion control family, the TCP-Reno is usually considered as base TCP. Since most of the congestion control algorithms work on the principle of TCP-Reno, almost all the algorithms have base procedures and functions as implemented in TCP-Reno with some amendment in an existing procedure or an addition of a new function. The standard functions that are implemented in the design of NS2's TCP-LP algorithm are mentioned below. Here only function body is presented, along with a brief note on the process that each function actually carries out.

```
tcp_lp_owd_calculator(...) {
//calculates One-Way Delay (OWD) in relative format. Original
  implement OWD as minus of remote time difference to local time
  difference directly. As this time difference, just simply
  equal to RTT, when the network status is stable, remote RTT
  will equal to local RTT, and result OWD into zero.
}



tcp_lp_rtt_sample (...) {
//calculate OWD, Record the min/max OWD and calculate SOWD.
}



tcp_lp_pkts_acked (...) {
//deal with active drop under early congestion indication.
 }



tcp_lp_remote_hz_estimator (...) {
//estimate remote HZ
}



tcp_lp_cong_avoid (...) {
//invoke only Reno congestion avoidance function when away from
  inference. As per TCP-LP paper, this will be handled in
  additive increase manner. This function is called every time
  an ACK is received and the cwnd can be increased.
  tcp_reno_cong_avoid (..)
}



tcp_lp_ssthresh(...) {
//this function is called when the TCP flow detects a loss. It
  returns the slow-start threshold of a flow after a packet loss
  is detected.
  tcp_reno_ssthresh (..)
}
```

The pseudo code shown below of TCP-LP congestion avoidance mechanism uses “*cwnd*” variable as the size of *cwnd* and the variable *“itti”* for interference time-out timer state indicator. The *itti* is set to 1 when the timer is initiated and to 0 when timer expires [78].

Pseudocode: TCP-LP Congestion Avoidance Policy

```
  new-ACK :    indication that ACK packet has arrived
  cong-ind:    congestion indication
  itti    :    inference time-out timer indication
  itt     :    inference timer period
  cwnd    :    congestion window

TCP-LP_CongestionAvoidance (…) {
// will only call new-Reno CA when away from inference
  if (new-ACK == 1) {
    if (cong-ind == 1) {
      if (itti == 1) {
          //drop cwnd into 1, within inference phase
          cwnd = 1;
    } else {
          // cwnd into half after the inference cut
          cwnd = cwnd/2;
      }
       //start of interference time
       itt = 1;
 } else {
       //after expiry of inference timer, additive
         increase of cwnd
  if (itti != 1) {
           cwnd += 1/cwnd;
    }
  }
 }
}
```

The NS2 implementation of TCP-LP uses the TCP Reno’s congestion avoidance mechanism for its normal congestion avoidance calls and will use the function tcp_reno_ssthresh for its normal *ssthresh* function. Since our intention is to tweak within *ssthresh* function, only that part is modified in the implementation.

6.3 Design & Implementation of Optimized vLP-TCP Algorithm

The original idea of TCP-LP is to utilize only the unused excess network bandwidth in a non-intrusive manner. The performance of TCP-LP will differ in different networks such as wired network and wireless network. Further, if TCP is used on VANET there arises the interesting question of how TCP application will influence or affect safety messaging flows when they compete for the shared bandwidth. Will they work in a friendly and cooperative way or in an aggressive and adversarial way? A simple idea is presented in this section to strike a balance and nudge the two contenders into improving their overall behaviour when TCP-LP coexists with safety messaging applications.

Guided by the understanding of the characteristics of various TCP congestion control variants and informed by the earlier rich experience in performance analysis conducted by the author, it has been decided to tweak TCP-LP slow-start mechanism in the same *ssthresh* function. The reason for tweaking only the slow-start phase of congestion control mechanism is described in detail in section 5.4. In the proposed vLP-TCP algorithm, the behaviour of slow-start is controlled in a way similar to that formulated for vScalable-TCP but with a small amendment in the Dropped Interval rate. This is done after making several trial and error observations on the behaviour of original TCP-LP.

As discussed earlier in the section of vScalable-TCP modelling, frequent calls to slow-start phase will invariably trigger more congestion under VANET scenario. The same phenomenon was observed during the experiments with original TCP-LP algorithm. Even though the throughput seems to be increasing due to the **slow-start** triggered by ***snd_ssthresh*** in the case of normal TCP-LP, actually the total number of dropped packets also gets increased a little. It means that some of the valuable bandwidth is being sacrificed for getting a little more throughput. For implementing the proposed idea, only the *ssthresh* function of vLP-TCP is modified as mentioned in Algorithm 2, while all the other functions of TCP-LP remain the same in this modified version of vLP-TCP.

Algorithm 2: Optimized vLP-TCP

```
 PreviousTime = 0
 DropInterval = 0
 MAXSHIFT     = 4
 MINSHIFT     = 2

tcp_vlp ssthresh(...)
   //here TCP-LP's default function tcp_reno_ssthresh(..) is
     replaced with improved version
 CurrentTime = tcp_time_stamp/1000.0;
 DropInterval = CurrentTime - PreviousTime;
if DropInterval <= 5 then
   //this will temporarily suspend slow-start until next
     restart
      snd_ssthresh = -1;
 else if DropInterval <=10 then
   //reduce much
      snd_ssthresh = tp -> snd_cwnd >> MAXSHIFT;
 else
   //reduce little
      snd_ssthresh = tp->snd_cwnd >> MINSHIFT;
 PreviousTime = CurrentTime;
 return snd_ssthresh
end of function
```

According to the proposed model, if the packet dropping due to congestion is high, then the function *ssthresh* will be called very frequently, so that the *DropInterval* will approach near 5 in value. At this point, the TCP slow-start is temporarily suspended until the next restart to avoid aggressive slow-start at the very congested situation. If the drop interval is above 5 and less than or equal to 10, then the *snd_ssthresh* is reduced much; otherwise, *snd_ssthresh* is reduced less. This maximum and minimum value can be selected based on how TCP flows are used in a practical application. Therefore, in the proposed vLP-TCP, the parameters of new algorithm are fixed based on the observation of the behaviour of normal TCP-LP under freeway sparse VANET scenario.

6.4 The Performance of Non-safety Messaging Application over vLP-TCP

This section presents the simulation results of the vLP-TCP performance for non-safety application in the freeway scenario. Though TCP-LP is designed to provide transparency to other TCP flows, now one has to deal with beacon based traffic instead. Similar to all the previous studies, a main safety short messaging application is periodically broadcasting as cross traffic during the TCP based file transfer flow. Scenario-2, with its associated parameters and the four metrics as discussed and defined in Chapter 3, applies equally well for the present study of TCP-LP and vLP-TCP performance. A few random vehicles in the scenario are accessing FTP resources from RSU in a multi-hop fashion, while at the same time all the 50 vehicles are periodically broadcasting safety services at varying intervals.

All the vehicles in the scenario are employed with telematics devices which use IEEE 802.11p MAC and PHY as presented in Chapter 2 and each vehicle must sense the state of channel during an AFIS before sending data. Since the data traffic is comprised of periodic BSM broadcast and FTP over TCP, the broadcast based traffic occupies bandwidth based on message interval rate whereas TCP based traffic does so based on *cwnd* dynamics. The lines on the graphs in this section denote the performance of TCP file transfer application.

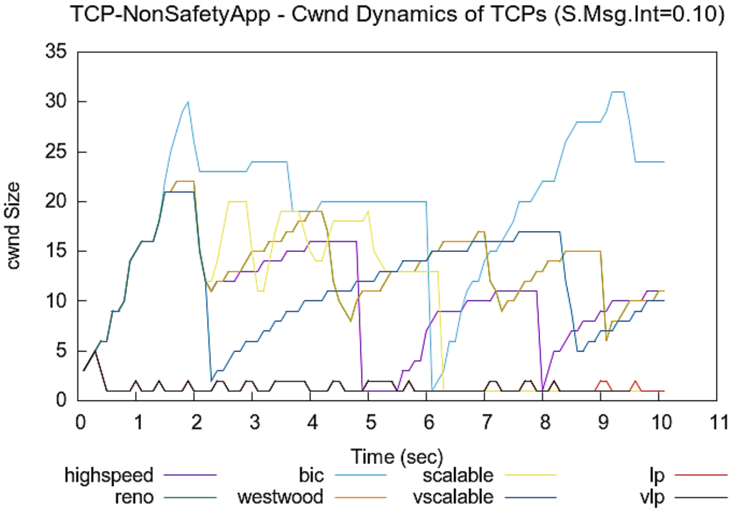


Figure 6.1: cwnd dynamics of TCP variants

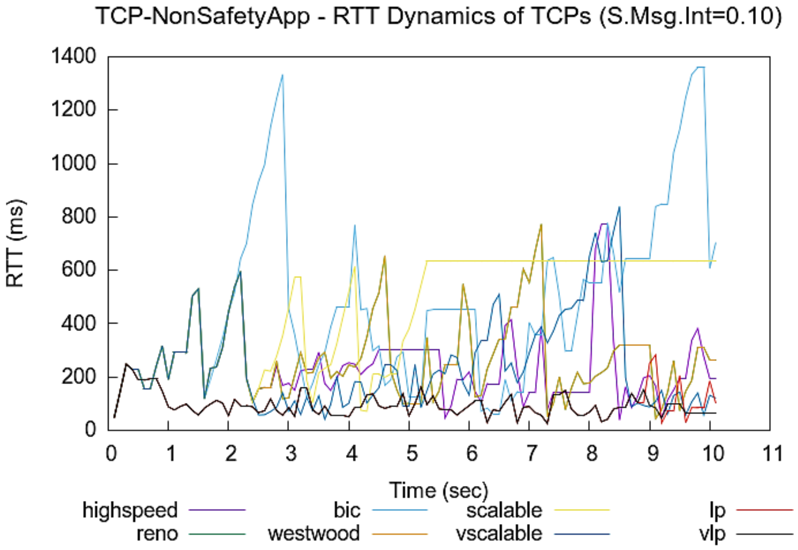


Figure 6.2: RTT dynamics of TCP variants

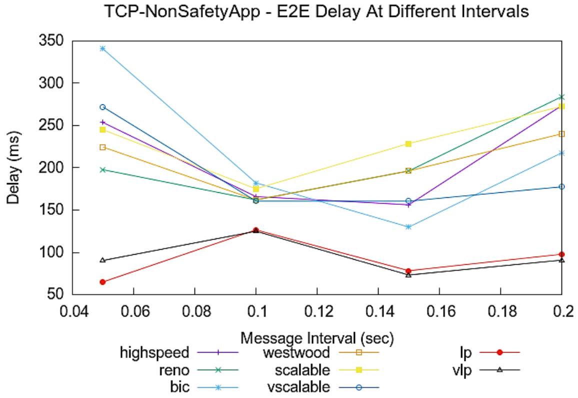


Figure 6.3: E2ED of TCP variants *vs* BSM intervals

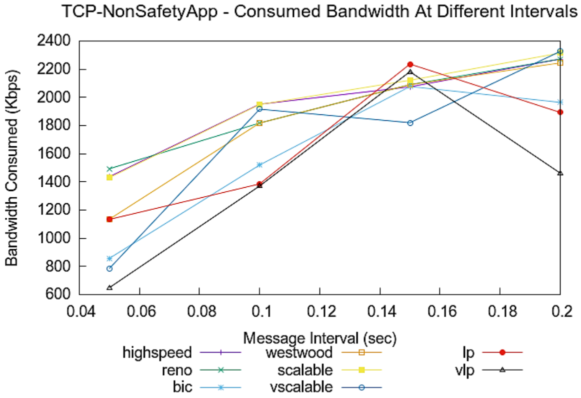


Figure 6.4: Average Throughput of TCP variants *vs* BSM intervals

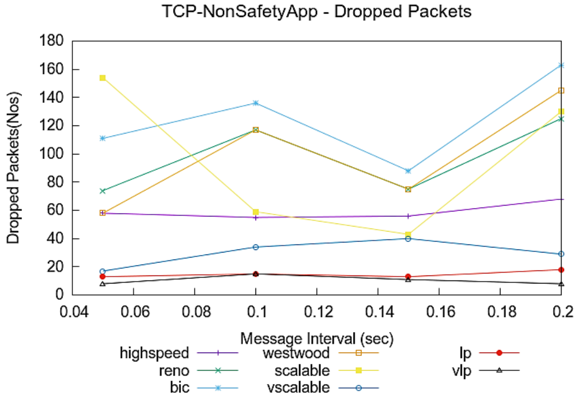


Figure 6.5: Dropped Packets of TCP variants *vs* BSM intervals

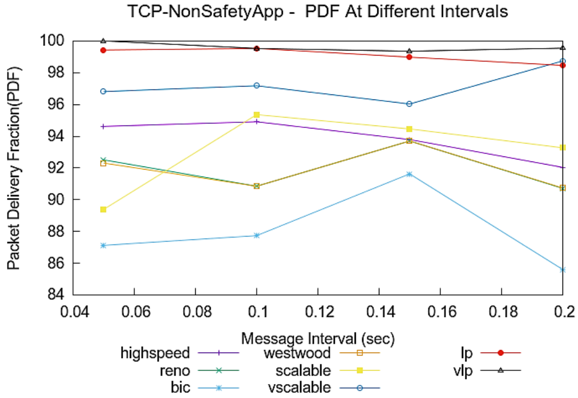


Figure 6.6: PDF of TCP variants *vs* BSM intervals

Figure 6.1 and Figure 6.2 show the *cwnd* and RTT dynamics respectively of different TCP algorithms, while running the FTP as a non-safety application for 10 seconds of simulation duration. Using PBC agent, the main beacon based application runs in the background, which is broadcasting safety messages at the interval of 0.10 seconds. The *cwnd* dynamics from this short duration of graph looks amorphous and deviates from the expected standard saw tooth pattern, yet remains vaguely interpretable. It is evident from the graph that the performance of the proposed vLP-TCP exhibits anomalous dynamics with the uniform least growth in comparison to all the other simulated algorithms.

Low window growth from this *cwnd* graph does not mean overall average low throughput will always be achieved, because this *cwnd* dynamics is the result of only one TCP traffic flow but not the average of multiple flows in the defined scenario. And it is generally known that the TCP's *cwnd* dynamic graphs created in VANET are not always reliable because of highly dynamic wireless channel conditions and it does not reflect true protocol behaviour as one encounter in a wired network. So, some degree of discrepancy in the corresponding average throughput graph is very likely and not unusual. Similarly, the RTT dynamics of single TCP flow is in correspondence with the variation of the *cwnd* growth. The purpose of *cwnd* and RTT graphs here is just to show a generalized performance curve in a very abstract manner. So, drawing any immediate conclusion of protocol behaviour from only these *cwnd* growth as well as RTT graph can never be conclusive and even may be misleading or risky.

Figure 6.3 and Figure 6.4 show respectively the performance in terms of average E2ED and throughput, achieved by the TCP non-safety file transfer application with respect to delay-sensitive safety message intervals rate. In comparison to all simulated TCPs, the TCP-LP and the proposed vLP-TCP show better performance in terms of E2ED metric with average low latency, right from 0.5s till 0.2s of message interval rate. The average 100ms of latency provided by vLP-TCP is just enough to support interactive infotainment applications that require faster response time. An important factor that contributes to lowest E2ED for both TCP-LP and vLP-TCP is due to their inherent non-intrusive behaviour.

Though there is least E2ED from both TCP-LP and vLP-TCP, the average consumed bandwidth of both the algorithms at the message interval of 0.10s is not as good as in other TCP variants. The reason for the under-performance in throughput of both TCP-LP and vLP-TCP is the non-intrusive nature of protocol that utilizes only unused excess bandwidth. In comparison to TCP-LP and other algorithms used here, the modified vLP-TCP at the message interval of 0.05s produces remarkably low throughput. With respect to TCP-LP, the 50% reduction of consumed bandwidth from vLP-TCP at the message interval rate of 0.05s is a direct consequence of the temporary freezing of slow-start mechanism that kicks in whenever packets are dropped excessively. Since the rate of *cwnd* growth of TCP-LP and vLP-TCP is based on low speed network and bandwidth is non-intrusively occupied, one cannot expect throughput similar to TCP-Reno and TCP-Scalable.

The performance of vScalable-TCP and vLP-TCP algorithms in terms of packets dropped and PDF is shown in Figure 6.5 and Figure 6.6 respectively. As seen in this graph, vLP-TCP obviously dropped less packets than vScalable-TCP, TCP-LP and all other compared algorithms. The fewer packets dropped from vLP-TCP is due to the *benevolent* nature of the proposed algorithm. It simply avoids packet dropping by regulating the TCP flow by invoking new *ssthresh* function. As stated earlier, the packet dropping is one-to-one related to PDF. That the packet drop is inversely proportional to PDF is evident from the line chart. The PDF is excellent in the case of vLP-TCP because it avoided packet drops by implementing the new slow-start freezing mechanism.

6.5 A Study on the Impact of vLP-TCP on Periodic BSM

The performance of vLP-TCP using various performance analysis metrics is evaluated in the previous section. Performance improvements in terms of E2ED, packet dropped rate and packet delivery ratio are achieved only by sacrificing average TCP throughput, specifically at the message interval rate of 0.05s. Vehicular safety being the core concern of DSRC protocol, it is necessary to evaluate the impact of modified vLP-TCP on BSM application so that the real characteristics of vLP-TCP and its applicability for VANET can be reliably estimated and suitably assessed.

Similar to the previous study, the main BSM application that is periodically broadcasting is kept as high priority traffic in the network, whereas a TCP based file transfer application is running as a low priority data flow. Scenario-2 with its associated parameters and the four performance analysis metrics as described in Chapter 3 forms the basis for this study as well.

The simulation results bring out the performance of BSM application with respect to TCP file transfer application in a freeway scenario. Though the scope is to compare the impact of TCP-LP *vis-à-vis* the proposed vLP-TCP, some other TCP algorithms are also included to get an overall idea about comparative performance improvement in the new algorithm. In the current scenario, all the 50 vehicles are periodically broadcasting safety services at varying intervals while at the same time a few vehicles are accessing FTP resources from a BS in a multi-hop fashion. Since the data traffic is comprised of beacon based BSM and FTP over TCP that are coexisting over same channel, again, as was the case before, the BSM broadcast data traffic is assumed to occupy the bandwidth based on message broadcast interval rate, whereas for TCP traffic it is based on choice of TCP's congestion window dynamics.

In this section, the simulation results are presented on the impact of vLP-TCP on the BSM application using file transfer application at different broadcast interval rates. The results actually show the performance of the BSM application with respect to the different TCP protocols used in TCP file transfer application. These graphs show how much the vLP-TCP flows affect the safety messaging application. The graphs also include WithoutTCP curve for the purpose of comparison when there is no of TCP flows.

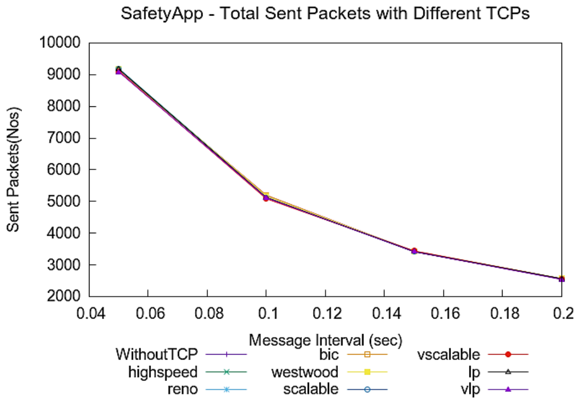


Figure 6.7: BSM Sent Packets in the presence of TCP variants

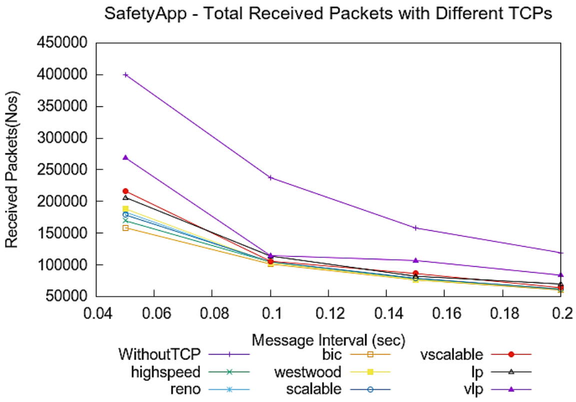


Figure 6.8: BSM Received Packets in the presence of TCP variants

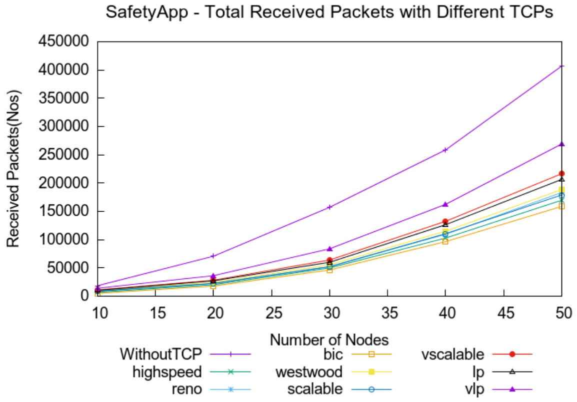


Figure 6.9: BSM Received Packets in the presence of TCP variants and No. of vehicles

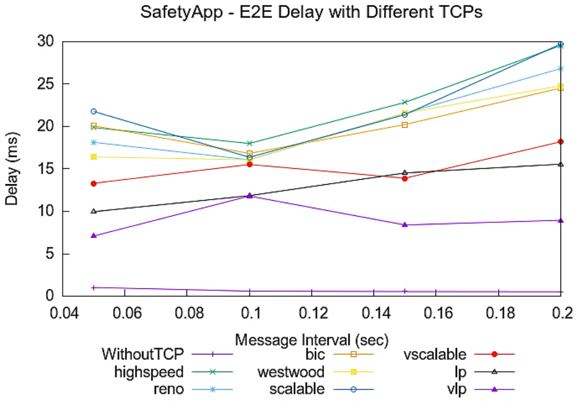


Figure 6.10: E2ED of BSM in the presence of TCP variants

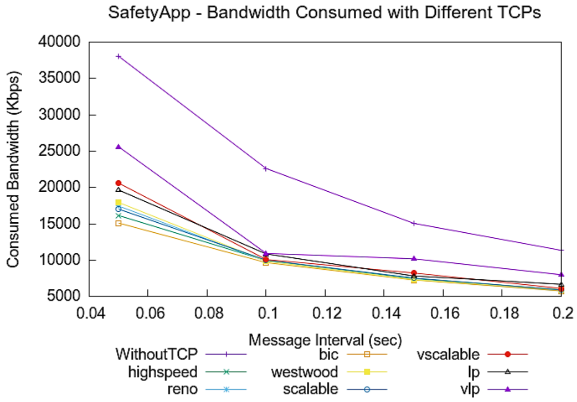


Figure 6.11: Average Throughput of BSM in the presence of TCP variants

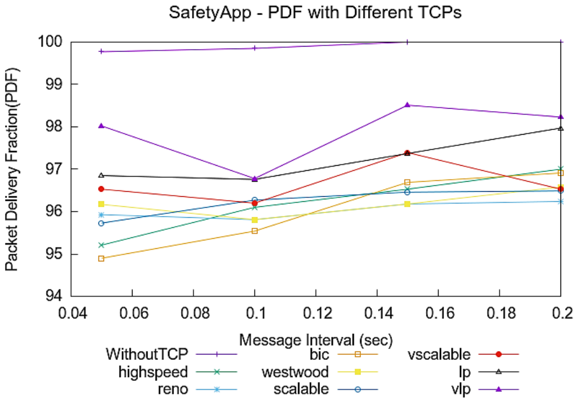


Figure 6.12: PDF of BSM in the presence of TCP variants

Figure 6.7 and Figure 6.8 show the performance in terms of total sent and total received packet respectively, achieved by the BSM at different broadcast interval rate and with different TCP flows. As shown in the graph, the send performance of the messaging application is not at all affected by any kind of TCP flows because safety messaging uses simple broadcast mechanism. So, it will just periodically send safety messages without waiting for ACK. However, the received message count is affected by all kinds of TCP flows and it was considerably different at lower message interval of the safety application. As shown in packet received graph, in the presence of vLP-TCP, the safety messaging application receives higher number of packets than any other algorithms at the message interval of 0.05s. The received packet count from Figure 6.9 shows that much superior flow of BSM is obtained in the presence of vLP-TCP and it is unaffected by the increasing number of vehicles, which is an appealing feature in the context of VANET.

As shown in the Figure 6.10, in comparison to WithoutTCP flow, the E2ED result is affected by all kinds of TCP traffic and it is considerably much affected at higher message interval of the safety application. Despite much affect, it is within the required average delay of 0.1s. In the presence of vLP-TCP, the E2ED of BSM application is much lower and least affected among other simulated TCPs. It signifies that the vLP-TCP does not affect much the safety messaging flow like the other TCP algorithms. Whereas consumed bandwidth as shown in Figure 6.11 is affected by all kinds of TCP flows, it is considerably much affected at higher interval of the safety application.

In the presence of vLP-TCP, the throughput of BSM is much better than any other algorithm, and this indicates the favourable effect of suppression of slow-start in the event of high packet drop. In the presence of vLP-TCP, the BSM application at the message interval of 0.05s achieves much higher throughput than any other TCP variants. Similarly, the PDF of vLP-TCP as shown in Figure 6.12, has considerably less impact on BSM at lower and higher broadcast intervals.

Encouraged by the improved performance result that is substantiated by the present simulation study, in terms of relevant metrics, it can safely and confidently be claimed that the new optimized vLP-TCP algorithm behaves, in comparison with other simulated TCP variants, in a friendly and cooperative manner in favour of BSM applications, which is the primary goal of this research work.

6.6 Chapter Summary

This work has successfully implemented the modelling idea formulated for vScalable-TCP and applied it to TCP-LP with some modification to arrive at the new algorithm vLP-TCP. The main purpose of applying the same optimization technique in vLP-TCP is to further validate the effectiveness of the proposed approach that is used in vScalable-TCP.

One cannot expect to obtain higher TCP throughput from TCP-LP than any other simulated algorithm, because TCP-LP utilizes only unused excess bandwidth in a non-intrusive manner. It is evident from the throughput results obtained that the performance of non-safety file transfer application using unmodified TCP-LP and modified vLP-TCP is along expected lines when the main *player* namely the broadcast based traffic, occupies the bandwidth based on beacon rate. However, the performance of FTP from vLP-TCP in terms of E2ED, PDF and packets dropped rate is far superior in comparison with other TCP variants.

Because of the intrinsic nature of the algorithm, both the TCP-LP and the proposed vLP-TCP performed well without affecting much the safety messaging applications. However, the modified vLP-TCP significantly improved the performance of the high priority safety applications in terms of throughput, PDF, and E2ED. But it should be noted that for both the vScalable-TCP and vLP-TCP algorithms the timely suppression of slow-start does reduce the TCP throughput; nevertheless, and perhaps fortuitously, it increases the overall performance of safety messaging applications which is indeed encouraging from the standpoint of VANET.

Chapter 7

Conclusion and Future Work

7.1 Introduction

In the last few years, VANETs research community have started paying increasing attention after the IEEE body introduced 802.11p communication protocol, which is a promising and dedicated technology for vehicular wireless network. The protocol design of DSRC/WAVE mainly targets time-bound safety messaging services but it also has features to co-support reliable non-safety infotainment type of use cases with the help of TCP. VANET, when using DSRC protocol suite, establishes V2V or V2I network spontaneously with a range of data transfer speeds. The V2V communication mode mainly helps in preventing accidents by using safety-critical applications, whereas V2I provides for a range of utility kind of non-safety services from roadside infostations. That TCP under performs in both MANET and VANET environments is well known and the early studies to ameliorate this situation did not take into account performance implications as to how far TCP based non-safety applications go well along with the main BSM application. This thesis has made performance evaluation as a focal point of study by giving due consideration to the essential metrics of TCP variants when used in the presence of sensitive safety short messaging traffic.

A detailed study of performance of TCP based file transfer application suggested that the ramification of tweaking *ssthresh* function of TCP's congestion control mechanism while accommodating safety messaging application should be comprehensively investigated. Such a study, which forms the major thrust of this research work has demonstrated the possibility that such tweaking of (any) TCP congestion control mechanism could indeed yield an overall better BSM performance in the presence of non-safety applications. Further, the present work has shown along the way that without tinkering TCP stack or without making existing protocol complex, TCP can be nudged to suitably fit and comfortably sit within the DSRC suite.

Since DSRC communication may require to have non-safety services co-exist with safety ones over the same channel, this research work has examined the feasibility of deploying TCP by taking into account both the contenders for the same channel. This effort has led to the optimization at the transport layer that made efficient use of the DSRC protocol, increasing the overall performance of BSM as well, but with some sacrifice of TCP throughput that was inevitable and only to be expected. In this research work, much attention has been given, as it should be, to the DSRC's core safety messaging application without compromising its integrity.

7.2 Summary of Results and Contributions

This thesis on the whole has focused on the feasibility of co-deploying both safety BSM message applications and non-safety file transfer applications in VANET, and in this endeavour has introduced a new technique in the TCP congestion control mechanism so that both categories of applications can coexist harmoniously in VANET environment. The results arrived at and the contributions made may be summarised, in step with the progression of this research study, as follows:

- The first part of this research has examined the performance of safety short messaging application in VANET environment. A simulation trace analysis is conducted over freeway vehicular traffic environment using IEEE 802.11p communication protocol. That analysis and the performance evaluation results indicate that the low index modulation scheme BPSK having data transfer rate of 3 Mbps is the most suitable one for safety short messaging application, even when the number of vehicles is scaled up and the distance between them gets longer. It is also noteworthy that with the increase of vehicles the throughput performance of BSM proportionally increases. Further, the simulation experiments show that the message broadcast interval rate of 0.05s or 0.10s works in favour of the strict time-bound safety system, even when the number of vehicles increases without bound.

- The simulation performance of BSM applications in the presence of non-safety use cases showed that it is very sensitive to the modulation schemes and the message broadcast interval. When the unused bandwidth of the channel is put to use for non-safety infotainment purposes, the plain TCP flows as well as the main BSM traffic suffer a serious setback. In view of these observations, the next part of the dissertation has examined the behaviour of many TCP variants with respect to BSM broadcasting.

- A separate simulation study of five TCP variants *viz.* TCP-Reno, TCP-BIC, TCP-Westwood, TCP-Highspeed and TCP-Scalable was conducted. When used in conjunction with BSM traffic, the performance of TCP versions varied greatly, presumably due to complex interaction between congestion control dynamics and message interval rates. An overall assessment revealed that TCP-Highspeed and TCP-Scalable behave in a stable and optimum way with respect to key metrics. This part of the work suggests the possibility of using TCP based services co-existing in a friendly manner with BSM traffic over DSRC channel, while keeping in check at the same time the likely negative impact on high priority safety message flows.

- In the next phase of the research, the impact of TCP flow over BSM application was evaluated and it is found unsurprisingly that BSM performance suffered a great deal and degraded drastically to varying degrees. Therefore, it became quite clear that the goal now should be to devise ways and means of ensuring unhindered flow of high priority BSM traffic while co-deploying infotainment services.

- To accomplish this goal stated just above, a novel technique is proposed to the TCP slow-start mechanism to handle congested situation. In this new approach, at highly congested situations, the slow-start is throttled temporarily to avoid further congestion and further packet drop. This idea has been successfully implemented on and integrated with TCP-Scalable and thereby a new algorithm named vScalable-TCP is derived. This new variant vScalable-TCP performed well without much affecting the safety messaging flows. In comparison with other simulated algorithms, vScalable-TCP provided the best file transfer performance but it did so at the cost of average throughput. Further and more importantly, in the presence of file transfer traffic, the vScalable-TCP significantly improved the performance of high priority safety applications in terms of PDF, E2ED and throughput and this is indeed an encouraging result.

- The final part of the dissertation carried further and successfully implemented a similar idea but with a few modifications in the packet drop interval rate on TCP-LP and a new algorithm called vLP-TCP is thus arrived at. This new proposed variant vLP-TCP too performed well without much affecting the safety messaging application. It provided the best performance but with some sacrifice of TCP throughput. Further, in comparison with all simulated TCPs including vScalable-TCP, the vLP-TCP significantly improved the performance of the high priority safety application in terms of PDF, E2ED and throughput. These results clearly prove the efficacy and indeed the potency of TCP’s slow-start suppression technique in the face of huge packet losses. Finally, it is observed that for both of the optimized vScalable-TCP and vLP-TCP algorithms, though the temporary freezing of slow-start does reduce TCP throughput to some extent, it nevertheless increases the overall performance of safety messaging application, which is a happy fallout of this novel optimization technique. And one may in fact intuitively expect this to be so.

7.3 Direction for Future Work

The overall results gathered from this research work open up new avenues and hint at some prospects for future research work, and a few of the remaining research issues may well serve as topics for further study. They are summarised below:

- The performance of non-safety file transfer application was evaluated using only single channel operation because of non-availability of p1609.4 in the NS2 simulator. DSRC protocol is designed to support 7 channels, where the central control channel CCH is dedicated for delay-sensitive safety data use, whereas service channel SCH is for non-safety uses cases. The switching from CCH to SCH at defined 50ms interval certainly has some effect on the performance of both safety and non-safety services. It would be of great interest to evaluate the TCP performance considering multi-channel operation in the VANET scenarios studied in this work.

- The performance study of high priority, periodic, broadcast-based safety messaging application was made, when low priority TCP file transfer application co-existed over the same channel. The support of EDCA (802.11e) on top of multi-channel operation makes the DSRC design theoretically robust for VANET deployment. When two different types of such services are running in DSRC protocol, safety messaging services should always be given high priority using EDCA of IEEE 802.11e mechanism. It would be of some interest to make a more realistic 802.11p based VANET simulation, if fully functional EDCA tool is embedded in DSRC protocol stack in NS2 to support QoS.
- The freeway mobility model has been used exclusively in this thesis to simulate vehicular mobility and to study its performance on periodic, broadcast-based safety and file transfer application. There are several other mobility models that can be used in urban and rural environments where mobility patterns are different, speed is much slower than freeway, and unlike arterial roads, wireless connections are somewhat stable. Low vehicular speed in city or rural roads will produce different sets of results for the same simulation scenarios. A recommended future research work could be to examine both the performance of beacon based safety message and the behaviour of TCP with the other mobility scenarios.
- Finally, at various places in the course of this thesis observations were made with regard bandwidth sharing between BSM and TCP-based applications. These evaluations and comparisons were made based on the simulation outputs

shown by a variety of graphs, wherein many protocol parameters and experimental factors come into play and interact in some *‘mysteriously’* complex and intricate way in controlling the access to spectrum. The superior performance of the two new congestion control algorithms - which indeed form the capstone of this research work – are corroborated by the simulation results, and discussions and comparative statements made in this context were invariably couched in relatively qualitative terms. In the spirit of scientific objectivity and further experimental credibility, this study can be extended by devising a way or methodology to capture the fairness property and to quantify this, perhaps by using some variation or modification of fairness equation such as Jain’s fairness index [165]. Or one may proceed *ab initio* by identifying first the contributing parameters and main ‘players’, and by suitably integrating or weighting their contributions to this feature. This would indeed be another way of extending the present work and might be a worthwhile pursuit.

*****

References

[1] P. Keeratiwintakorn, E. Thepnorarat, and A. Russameesawang, “Ubiquitous communication for V2V and V2I for Thailand intelligent transportation system,” in: Proceedings of NTC International Conference, Thailand, 2009.

[2] S. An, B.-H. Lee, and D.-R. Shin, “A Survey of Intelligent Transportation Systems,” in: Proceedings of 3rd International conference on Computational Intelligence, Communication Systems and Networks, July 2011, pp. 332–337.

[3] Y. L. Morgan, “Managing DSRC and WAVE Standards Operations in a V2V Scenario,” International Journal of Vehicular Technology, 2010.

[4] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, “A comprehensive survey on vehicular Ad Hoc network,” Journal of Network and Computer Application, vol. 37 (2014), pp. 380–392.

[5] S. Yousefi, T. Chahed, S. M. M. Langari, and K. Zayer, “Comfort applications in vehicular ad hoc networks based on fountain coding,” in: Proceedings of 71st IEEE Vehicular Technology Conference, 2010, pp. 1–5.

[6] M. A. Abakar, R. A. Saeed, A. A. Hassan, O. M. Mohammed, O. Khalifa, and S. Islam, “The challenges of wireless internet access in vehicular environments,” in: Proceedings of International Conference on Information and Communication Technology for the Muslim World, 2010, pp. D31–D36.

[7] X. Wu *et al.*, “Vehicular Communications Using DSRC: Challenges, Enhancements, and Evolution,” IEEE Journal on Selected Areas in Communication., vol. 31, no. 9 (Sep. 2013), pp. 399–408.

[8] J. B. Kenney, “Dedicated Short-Range Communications (DSRC) Standards in the United States,” In: Proceedings of the IEEE, vol. 99, no. 7, Jul. 2011, pp. 1162–1182.

[9] H. T. Cheng, H. Shan, and W. Zhuang, “Infotainment and road safety service support in vehicular networking: From a communication perspective,” Mechanic. Systems and Signal Processing, vol. 25, no. 6 (2011), pp. 2020–2038.

[10] Y. Qian and N. Moayeri, “Design of secure and application-oriented VANETs,” in: Proceedings of IEEE Vehicular Technology Conference, 2008, pp. 2794–2799.

[11] Z. Ahmed and F. Bai, “Vehicle Safety Communications – Applications (VSC-A) Final Report,” Department of Transportation, Washington, DC 20590, DOT HS 811 492C, Sep. 2011.

[12] M. Schulze, “PReVENT,” European Commission - Information Society and Media DG, Feb-2004.

[13] M. Rosenquist, “Cooperative Vehicle- Infrastructure Systems (CVIS),” ERTICO – ITS Europe, SP 1.3, 2010.

[14] A. Frötscher, “Co-operative Networks for Intelligent Road Safety(COOPERS),” AustriaTech, Austria, Draft D16-IR 8200/8500, 2010.

[15] A. Festag, G. Noecker, and M. Strassberger, “NoW – Network on Wheels’: Project Objectives, Technology and Achievements,” in: Proceedings of 5th International Workshop on Intelligent Transportation System (WIT), Germany, 2008, pp. 211–216.

[16] W. Enkelmann, “FleetNet - applications for inter-vehicle communication,” in: Proceedings of the IEEE Intelligent Vehicles Symposium, (Cat. No.03TH8683), 2003, pp. 162–167.

[17] M. S. Anwer and C. Guy, “A Survey of VANET Technologies,” Journal of Emerging Trends Computer Information Science, vol. 5, no. 9 (2014) pp. 661–671.

[18] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervello-Pastor, “From Delay-Tolerant Networks to Vehicular Delay-Tolerant Networks,” IEEE Communication Survey Tutorial., vol. 14, no. 4 (2012) pp. 1166–1182.

[19] M. Sichitiu and M. Kihl, “Inter-vehicle communication systems: a survey,” IEEE Communication Survey Tutorial, vol. 10, no. 2 (2008), pp. 88–105.

[20] J. Ott and D. Kutscher, “The Drive-thru’ architecture: WLAN-based internet access on the road,” in: Proceedings of 59th IEEE Vehicular Technology Conference, vol. 5, 2004, pp. 2615–2622.

[21] J. Ott and D. Kutscher, “Internet access for mobile users: From drive-thru internet to delay-tolerant ad-hoc networking,” Multi-Hop Ad Hoc Network from Theory to Reality, p. 241-277, Nova Science Publisher, Inc, 2007.

[22] S. Ezell, “Intelligent Transportation Systems,” The Information Technology & Innovation Foundation (ITIF), Washington, DC, USA, 2010.

[23] G. Dimitrakopoulos, “Intelligent transportation systems based on internet-connected vehicles: Fundamental research areas and challenges,” in: Proceedings of the 11th International Conference on ITS Telecommunications, St.Petersburg, 2011, pp. 145–151.

[24] S. Yousefi, M. S. Mousavi, and M. Fathy, “Vehicular ad hoc networks (VANETs): challenges and perspectives,” in: Proceedings of 6th IEEE International Conference on ITS Telecommunications, 2006, pp. 761–766.

[25] R. Stewart and C. Metz, “SCTP: new transport protocol for TCP/IP,” IEEE Internet Computing, vol. 5, no. 6 (2001), pp. 64–69.

[26] B. Yu and F. Bai, “ETP: Encounter Transfer Protocol for opportunistic vehicle communication,” IEEE INFOCOM (2011), pp. 2201–2209.

[27] J. Eriksson, H. Balakrishnan, and S. Madden, “Cabernet: vehicular content delivery using WiFi,” in: Proceedings of 14th ACM International Conference on Mobile Computing and Networking, 2008, pp. 199–210.

[28] C. Christopher, R. M. Noor, and E. Mostajeran, “Optimizing wireless channel using adaptive modulation to improve QoS in VANET,” Malaysian Journal of Computer Science, vol. 26, no. 1 (2013), pp. 34–43.

[29] R. Nere and U. Nagaraj, “Scrutiny of Broadcasting Protocols in VANET,” International Journal of Grid Computing Application, vol. 6, no. 1/2 (2015), pp. 13–25

[30] R.-S. Chang and H.-C. Ting, “Improving the performance of broadcasting in ad hoc wireless networks,” in: Proceedings of 8th International Conference on Parallel and Distributed Systems (ICPADS), 2001, pp. 69–74.

[31] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” IEEE Transaction on Information Theory, vol. 46, no. 2 (2000), pp. 388–404.

[32] Jinyang Li, Charles Blake, Douglas S. J. De Couto, Hu Imm Lee, and Robert Morris, “Capacity of Ad Hoc Wireless Networks,” in: Proceedings of 7th International Conference on Mobile Computing and Networking, 2001, pp.61-69.

[33] Z. Wang and M. Hassan, “How much of dsrc is available for non-safety use?,” in: Proceedings of the 5th ACM International Workshop on VehiculAr Inter-Networking, 2008, pp. 23–29.

[34] S. S. Anjum, R. M. Noor, and M. H. Anisi, “Survey on MANET Based Communication Scenarios for Search and Rescue Operations,” in: Proceedings of the 5th International Conference on IT Convergence and Security (ICITCS), 2015, pp. 1–5.

[35] B. Aziz, E. Nourdine, and E. K. Mohamed, “A Recent Survey on Key Management Schemes in MANET,” in: Proceedings of 3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA), 2008, pp. 1–6.

[36] A. Srivastava, A. Mishra, B. Upadhyay, and A. k Yadav, “Survey and overview of Mobile Ad-Hoc Network routing protocols,” in: Proceedings of IEEE International Conference on Advances in Engineering and Technology Research (ICAETR), 2014, pp. 1–6.

[37] A. B. Al-Khalil, A. Al-Sherbaz, and S. Turner, “Enhancing the Physical Layer in V2V Communication Using OFDM–MIMO Techniques,” in: Proceedings of 14th Annual Post Graduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, 2013, pp. 10.

[38] H. Hartenstein and K. P. Laberteaux, “A tutorial survey on vehicular ad hoc networks,” IEEE Communication Magazine, vol. 46, no. 6 (2008), pp. 164–171.

[39] J. Harri, F. Filali, and C. Bonnet, “Mobility models for vehicular ad hoc networks: a survey and taxonomy,” IEEE Communication Survey Tutorial, vol. 11, no. 4(2009), pp. 19–41.

[40] J. Jakubiak and Y. Koucheryavy, “State of the art and research challenges for VANETs,” in: Proceedings of 5th IEEE Consumer communications and networking conference (CCNC), 2008, pp. 912–916.

[41] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, “Progress and challenges in intelligent vehicle area networks,” Communication of the ACM, vol. 55, no. 2 (2012), p. 90.

[42] T. Kosch, I. Kulp, M. Bechler, M. Strassberger, B. Weyl, and R. Lasowski, “Communication architecture for cooperative systems in Europe,” IEEE Communication Magazine, vol. 47, no. 5 (2009), pp. 116–125.

[43] Q. Zhao, Y. Zhu, C. Chen, H. Zhu, and B. Li, “When 3G Meets VANET: 3G-Assisted Data Delivery in VANETs,” IEEE Sensors Journal, vol. 13, no. 10 (2013), pp. 3575–3584.

[44] S. Panichpapiboon and W. Pattara-atikom, “A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks,” IEEE Communication Survey & Tutorial, vol. 14, no.3 (2011), pp. 784-196.

[45] P. Ranjan and K. K. Ahirwar, “Comparative Study of VANET and MANET Routing Protocols,” in: Proceedings of the International Conference on Advanced Computing and Communication Technologies, 2011, PP. 517-523.

[46] P. Tiwari and R. S. Kushwah, “Enhancement of VANET Communication Range using WiMAX and Wi-Fi: A Survey,” International Journal of Urban Design for Ubiquitous Computing, vol. l.1, no. 1(2013), pp. 11–18.

[47] F. Li and Y. Wang, “Routing in vehicular ad hoc networks: A survey,” IEEE Vehicular Technology Magazine, vol. 2, no. 2 (2007), pp. 12–22.

[48] K. Prasanth, D. K. Duraiswamy, K. Jayasudha, and D. C. Chandrasekar, “Improved packet forwarding approach in Vehicular ad hoc networks using RDGR algorithm,” International Journal of Computer Network, vol. 2, no. 1 (2010), pp. 1–15.

[49] I. T. S. Committee and others, “IEEE trial-use standard for wireless access in vehicular environments-security services for applications and management messages,” IEEE Vehicular Technology Society Standard, vol. 1609, 2006.

[50] IEEE Std. 802.11, “IEEE Standard for Information Technology-Telecommunications and Information Exchange between Systems-Local and Metropolitan Area Networks-Specific Requirements Part 11,” 2007. Retrieved 15-Jun-2015 from: http://www.ie.itcr.ac.cr/acotoc/Ingenieria/Lab%20TEM%20II/Antenas/Especificacion%20802%2011-2007.pdf

[51] IEEE P802.11p, “Amendment 3: wireless access in vehicular environments (WAVE)," Draft D0.26, 2006.

[52] IEEE P1609.1, “Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager," 2006.

[53] IEEE P1609.2, “Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Security Services for Applications and Managements Messages," 2006.

[54] IEEE P1609.3, “Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Networking Services," 2007.

[55] IEEE P1609.4, “Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation," 2006.

[56] A. Q. Fitzgerald and S. Robert, “Comments of the Alliance of Automobile Manufacturers Inc. and the Association of Global Automakers Inc.,” Before the Federal Communication Commission, Washington, DC-20554, 2013, Retrieved 11-Oct-2015 from: https://www.globalautomakers.org/system/files/document/attachments/Reply%20Comments%20-%205%20Ghz%20Spectrum.pdf

[57] M. van Eenennaam, A. van de Venis, and G. Karagiannis, “Impact of IEEE 1609.4 channel switching on the IEEE 802.11p beaconing performance,” in: Proceedings of the IEEE IFIP Wireless Days (WD), 2012, pp. 1–8.

[58] D. Acatauassu, I. Couto, P. Alves, and K. Dias, “Performance Evaluation of Inter-Vehicle Communications Based on the Proposed IEEE 802.11p Physical and MAC Layers Specifications,” in: Proceedings of 10th International Conference on Networks, 2011, pp. 170–174.

[59] Vehicular Technology Society, Intelligent Transportation Systems Committee, IEEE and IEEE-SA Standards Board, IEEE trial-use standard for WAVE multi-channel operation. 2006, Retrieved 20-Nov-2015 from: http://ieeexplore.ieee.org/servlet/opac?punumber=6956536

[60] L. Miao, K. Djouani, B. J. Wyk, Y. Hamam, “A survey of IEEE 802.11p MAC protocol,” Journal of Selected Areas in Telecommunication (JSAT), (2011), pp.1-7.

[61] K. Bilstrup, E. Uhlemann, E. G. Ström, and U. Bilstrup, “Evaluation of the IEEE 802.11p MAC method for vehicle-to-vehicle communication,” in: Proceedings of IEEE 68th Vehicular Technology Conference (VTC), 2008, pp. 1–5.

[62] T. Weil, “Service management for ITS using WAVE (1609.3) networking,” in: Proceedings of the IEEE GLOBECOM Workshops, 2009, pp. 1–6.

[63] D. Eckhoff and C. Sommer, “A Multi-Channel IEEE 1609.4 and 802.11p EDCA Model for the Veins Framework,” in: Proceedings of 5th ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems(SIMUTools): 5th ACM/ICST International Workshop on OMNeT++. ACM, 2012.

[64] Q. Chen, D. Jiang, and L. Delgrossi, “IEEE 1609.4 DSRC multi-channel operations and its implications on vehicle safety communications,” in: Proceedings of the IEEE Vehicular Networking Conference (VNC), 2009, pp. 1–8.

[65] Y. J. Li, “An overview of the DSRC/WAVE technology,” in Quality, Reliability, Security and Robustness in Heterogeneous Networks, Springer, 2012, pp. 544–558.

[66] M.-W. Li, T. Wu, W.-Y. Lin, K. Lan, C.-M. Chou, and C.-H. Hsu, “On the Feasibility of Using 802.11p for Communication of Electronic Toll Collection Systems,” In: Proceedings of the IEEE International Conference on Network-Based Information Systems, 2011, pp. 68–75.

[67] K. Bilstrup, “A survey regarding wireless communication standards intended for a high-speed vehicle environment,” Technical Report IDE 0712, Hamstad University, Sweden, 2007.

[68] F. Bai, D. D. Stancil, and H. Krishnan, “Toward understanding characteristics of dedicated short-range communications (DSRC) from a perspective of vehicular network engineers,” in: Proceedings of 16th Annual International Conference on Mobile Computing and Networking, 2010, pp. 329-340.

[69] Andrew Pickford, “Dedicated-Short-Range-Competition,” Retrieved 17-Jan-2016 from: http://www.ttc-global.com/uploads/Dedicated-Short-Range-Competition.pdf

[70] W. R. Stevens, TCP/IP Illustrated, vol. 1. Addison-Wesley, 1994.

[71] W. R. Stevens, M. Allman, and V. Paxson, “TCP Congestion Control,” Retrieved 07-Oct-2015 from: https://tools.ietf.org/html/rfc2581

[72] W. R. Stevens, “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms,” Retrieved 12-Oct-2015 from: https://tools.ietf.org/html/rfc2001

[73] V. Jacobson, “Berkeley TCP Evolution from 4.3-Tahoe to 4.3 Reno,” in: Proceedings of the 18th Internet Engineering Task Force, University of British Colombia, Vancouver, BC, 1990.

[74] L. Xu, K. Harfoush, and I. Rhee, “Binary increase congestion control (BIC) for fast long-distance networks,” in: Proceedings of 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM, 2004, vol. 4, pp. 2514–2524.

[75] T. Kelly, “Scalable TCP: Improving performance in highspeed wide area networks,” ACM SIGCOMM Computer Communication Review, vol. 33, no. 2 (2003), pp. 83–91.

[76] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi, and R. Wang, “TCP Westwood: end-to-end congestion control for wired/wireless networks,” Wireless Network, vol. 8, no. 5 (2002), pp. 467–479.

[77] S.Floyd, “HighSpeed TCP for large congestion windows.” IETF RFC 3649, 2003.

[78] A. Kuzmanovic and E. W. Knightly, “TCP-LP: low-priority service via end-point congestion control,” IEEE ACM Transaction on Networking, vol. 14, no. 4 (2006), pp. 739–752.

[79] X. Huang, C. Lin, F. Ren, and H. Yin, “Highspeed tcp modeling and analysis,” in: Proceedings of the PATHNets Workshop, 2004.

[80] M. Bechler, W. J. Franz, and L. Wolf, “Mobile internet access in FleetNet,” 13th Fachtagung. Kommunikation Verteilten Systemen Leipzing, 2003.

[81] G. Holland and N. Vaidya, “Analysis of TCP performance over mobile ad hoc networks,” Wireless Network, vol. 8, no. 2/3 (2002), pp. 275–288.

[82] A. Khan, S. Sadhu and M. Yeleswarapu, “A comparative analysis of DSRC and 802.11 over Vehicular Ad hoc Networks, Project Report, Department of Computer Science, University of California, Santa Barbara, 2009.

[83] P. Manzoni, D. Ghosal, and G. Serazzi, “A Simulation Study of the Impact of Mobility on TCP/IP,” in: Proceedings of the International Conference on Network Protocols (ICNP), Boston, 1994, pp. 196–203.

[84] R. Gass, J. Scott, and C. Diot, “Measurements of in-motion 802.11 networking,” in: Proceedings of the IEEE Workshop on Mobile Computing Systems & Applications (WMCSA), 2006, pp. 69–74.

[85] A. Roos *et al.*, “Broadband Wireless Internet Access in Public Transportation,” In: Proceedings of the VDE congress Innovations for Europe, Aachen, Germany, 2006.

[86] J. Eriksson, H. Balakrishnan, and S. Madden, “Cabernet: vehicular content delivery using WiFi,” in: Proceedings of 14th ACM International Conference on Mobile Computing and Networking (MobiCom), 2008, pp. 199–210.

[87] M. Bechler, S. Jaap, and L. Wolf, “An optimized tcp for internet access of vehicular ad hoc networks,” in: Proceedings of the IFIP Networking Conference, 2005.

[88] A. Chen, B. Khorashadi, D. Ghosal, and C.-N. Chuah, “Impact of transmission power on TCP performance in vehicular ad hoc networks,” in: Proceedings of 4th Annual Conference on Wireless on Demand Network Systems and Service (WONS), 2007, pp. 65–71.

[89] Z. Fu, X. Meng, and S. Lu, “How bad TCP can perform in mobile ad hoc networks,” in: Proceedings of 7th international symposium on Computers and communications (ISCC), 2002, pp. 298–303.

[90] A. Cardote, “Transport Protocol for Vehicular Environments: Challenges and Trends,” Electrónica E Telecomunicações, vol. 5, no. 2 (2010), pp. 202–207.

[91] “The European Telecommunications Standards Institute, ETSI TS 102 637-2 v1.2.1 - Intelligent Transport Systems (ITS) - Vehicular Communications - Basic Set of Applications - Part2: Specification of Cooperative Awareness Basic Service,” Technical Report, March 2011, Retrieved 9-April-2015 from: http://www.etsi.org/deliver/etsi_ts/102600_102699/10263702/01.02.01_60/ts_10263702v010201p.pdf

[92] “Dedicated Short Range Communications (DSRC) Message Set Dictionary- SAE J2735-201603.” SAE International, 2016, Retrieved 5-Jan-2016 from: http://standards.sae.org/j2735_201603/

[93] T. Ozan, W. Nawaporn, B. Fan, M. Priyantha, and S. Varsha, “Broadcasting in VANET,” in: Proceedings of the IEEE INFOCOM, 2008, pp. 7–12.

[94] Q. Chen, D. Jiang, V. Taliwal, and L. Delgrossi, “IEEE 802.11 based vehicular communication simulation design for NS-2,” in: Proceedings of 3rd International Workshop on Vehicular Ad Hoc Networks, ACM, 2006, pp. 50-56.

[95] VSCC. (2005), “Vehicle Safety Communications Project Task 3 Final Report: Identify Intelligent Vehicle Safety Applications Enabled by DSRC.” National Highway Traffic Safety Admin., Washington, DC, Technical Report, DOT HS 809 859, 2005.

[96] R. K. Schmidt, T. Leinmüller, E. Schoch, F. Kargl, and G. Schäfer, “Exploration of adaptive beaconing for efficient intervehicle safety communication,” *IEEE Network*, vol. 24, no. 1 (2010), pp. 14–19.

[97] C. Campolo and A. Molinaro, “Multichannel communications in vehicular Ad Hoc networks: a survey,” IEEE Communication Magazine, vol. 51, no. 5 (2013), pp. 158–169.

[98] Y. Chang, M. Lee, and J. A. Copeland, “An Adaptive On-Demand Channel Estimation for Vehicular Ad Hoc Networks,” in: Proceedings of 6th IEEE Consumer Communications and Networking Conference (CCNC), 2009, pp. 1–5.

[99] M. Hassan, M. El-Taruni, and R. Zrae, “A Joint Adaptive Modulation and Channel Coding Scheme for Multimedia Communications over Wireless Channels,” International Journal of Wireless and Mobile Computing, vol. 9, no. 1 (2015), pp. 99–114.

[100] S. C. K. Lye, S. E. Tan, Y. K. Chin, B. L. Chua, and K. T. K. Teo, “Performance analysis of intelligent transport systems (ITS) with adaptive transmission scheme,” in: Proceedings of 4th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2012, pp. 418–423.

[101] M. Javed and J. Y. Khan, others, “Performance analysis of an adaptive rate-range control algorithm for VANET safety applications,” in: Proceedings of International Conference on Computing, Networking and Communications (ICNC), 2014, pp. 418–423.

[102] Y. Park and H. Kim, “Application-Level Frequency Control of Periodic Safety Messages in the IEEE WAVE,” IEEE Transaction Vehicular Technolog*y*, vol. 61, no. 4 (2012), pp. 1854–1862.

[103] C.-L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan, “Adaptive intervehicle communication control for cooperative safety systems,” IEEE Network, vol. 24, no. 1 (2010), pp. 6–13.

[104] M. H. Rahman and M. Nasiruddin, “Impact of two realistic mobility models for vehicular safety applications,” in: Proceedings of the International Conference on Informatics, Electronics & Vision (ICIEV), 2014, pp. 1–6.

[105] S. Yousefi, M. Fathy, and A. Benslimane, “Performance of beacon safety message dissemination in Vehicular Ad hoc Networks (VANETs),” Journal of Zhejiang University Science (JZUS*)*, vol. 8, no. 12 (2007), pp. 1990–2004.

[106] S. Al-Khayatt and A. Dogman with A. Jafari, “Performance evaluation of IEEE 802.11p for vehicular communication networks,” in: Proceedings of 8th IEEE,

IET International Symposium on Communication Systems, Networks and Digital Signal Processing, 2012, pp. 1–5.

[107] L Yang, J. Guo, and Y. Wu, "Channel adaptive one hop broadcasting for vanets," in: Proceedings of 11th IEEE International Conference on Intelligent Transportation Systems (ITSC), 2008, pp. 369–374.

[108] K. Xu, S. Bae, S. Lee, and M. Gerla, "TCP behavior across multihop wireless networks and the wired internet," in: Proceedings of 5th ACM international workshop on Wireless mobile multimedia, 2002, pp. 41–48.

[109] E. Spaho, M. Ikeda, L. Barolli, V. Kolici, F. Xhafa, and M. Younas, "Investigation of TCP Traffic in a Vehicular Ad-hoc Network Considering DYMO Routing Protocol," in: Proceedings of 3rd International Conference on Emerging Intelligent Data and Web Technologies, 2012, pp. 111–116.

[110] E. Spaho, S. Sakomoto, M. Ikeda, L. Barolli, F. Xhafa, and V. Kolici, "Multiflow TCP Traffic in VANETs: Performance Comparison of OLSR and AODV Routing Protocols," in: Proceedings of 4th International Conference on Emerging Intelligent Data and Web Technologies, 2013, pp. 33–38.

[111] F. Pirmohammadi, M. Fathy, and H. Ghaffarian, "TCP and UDP Fairness in Vehicular Ad hoc Networks," International Journal of Emerging Technologies and Advance Engineering, vol. 2, no. 6 (2012), pp. 111-118

[112] A. Rahim, M. Sher, A. Javed, I. Ahmad, and R. Hameed, "Performance analysis of TCP in VANETs by using 802.11e," in: Proceedings of 16th IEEE International Conference on Networks (ICON), 2008, pp. 1–3.

[113] J.-L. Kuo, C.-H. Shih, and Y.-C. Chen, "Performance analysis of real-time streaming under TCP and UDP in VANET via OMNET," in: Proceedings of 13th International Conference on ITS Telecommunications (ITST), 2013, pp. 116–121.

[114] B. Paul, A. Roy, and S. K. Paul, "Comparison of DSR, AODV, and DSDV Routing Protocols with Varying Pause Time & Node Density over TCP & CBR Connections in VANET," in: Proceedings of 6th International Conference on Computational Intelligence and Communication Networks, 2014, pp. 374–379.

[115] J. A. Jude and R. Ganesan, "A comprehensive experimental analysis of standard TCP variants in vehicular environment," in: Proceedings of the International Conference on Computing and Communications Technologies (ICCCT), 2015, pp. 350–355.

[116] D. Jaiswal and S. S. Bhadauria, "Performance Analysis of UDP/CBR &TCP/FTP Traffic Under Reactive and Proactive Routing Protocols in VANET," International Research Journal of Engineering and Technology, vol.2 no.4 (2015), pp. 700-709.

[117] R. Kaur and G. S. Josan, "Performance Evaluation of Congestion Control Tcp Variants in Vanet Using Omnet+," International Journal of Engineering Research and Applications (IJERA), vol. 2, no. 5 (2012), pp. 1682–1688.

[118] K. Dalal, P. Chaudhary, and P. Dahiya, “Performance Evaluation of TCP and UDP Protocols in VANET Scenarios using NCTUns-6.0 Simulation Tool,” International Journal of Computer Application, vol. 36, no. 6 (2011), pp. 6-9.

[119] M. Al-Hasanat, K. Seman, and K. Saadan, “Enhanced TCPW’s fast retransmission and fast recovery mechanism over high bit errors networks,” in: Proceedings of the International Conference on Computer, Communications, and Control Technology (I4CT), 2015, pp. 337–340.

[120] S. Henna, “A Throughput Analysis of TCP Variants in Mobile Wireless Networks,” in: Proceedings of 3rd International Conference on Next Generation Mobile Applications, Services and Technologies, 2009, pp. 279–284.

[121] B. Hadrien and R. Les Cottrell, “Evaluation of Advanced TCP Stacks on Fast Long-Distance Production Network,” in: Proceedings of 1st International Workshop on protocols for Fast Long-Distance Networks, 2004.

[122] A. T. Giang and A. Busson, “Modeling CSMA/CA in VANET”, Analytical and Stochastic Modeling Techniques and Applications,” Lecture Notes in Computer Science, 2012, pp. 91-105, Retrieved 15-May-2015 from: https://link.springer.com/chapter/10.1007%2F978-3-642-30782-9_7

[123] G. P. Grau, D. Pusceddu, S. Rea, O. Brickley, M. Koubek, and D. Pesch, “Vehicle-2-vehicle communication channel evaluation using the CVIS platform,” in: Proceedings of the 7th International Symposium on Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010, pp. 449–453.

[124] B. E. Bilgin and V. C. Gungor, “Performance Comparison of IEEE 802.11p and IEEE 802.11b for Vehicle-to-Vehicle Communications in Highway, Rural, and Urban Areas,” International Journal of Vehicular Technology, vol. 2013, pp. 1–10, 2013.

[125] J. Gozálvez, M. Sepulcre, and R. Bauza, “IEEE 802.11 p vehicle to infrastructure communications in urban environments,” IEEE Communication Magazine, vol. 50, no. 5 (2012), pp. 176–183.

[126] C. Sommer, D. Eckhoff, R. German, and F. Dressler, “A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments,” in: Proceedings of 8th International Conference on Wireless On-Demand Network Systems and Services, 2011, pp. 84–90.

[127] J. C. Burguillo-Rial, E. Costa-Montenegro, F. Gil-Castineira, and P. Rodríguez-Hernández, “Performance analysis of IEEE 802.11 p in urban environments using a multi-agent model,” in: Proceedings of 19th IEEE International Symposium on, Personal, Indoor and Mobile Radio Communications (PIMRC), 2008, pp. 1–6.

[128] J.-M. Chung, M. Kim, Y.-S. Park, M. Choi, S. Lee, and H. S. Oh, “Time Coordinated V2I Communications and Handover for WAVE Networks,” IEEE Journal on Selected Areas in Communication, vol. 29, no. 3 (2011), pp. 545–558.

[129] Y. Yao, L. Rao, and X. Liu, “Performance and Reliability Analysis of IEEE 802.11p Safety Communication in a Highway Environment,” IEEE Transaction Vehicular Technology, vol. 62, no. 9 (2013), pp. 4198–4212.

[130] C. Han, M. Dianati, R. Tafazolli, and R. Kernchen, “Throughput analysis of the IEEE 802.11 p enhanced distributed channel access function in vehicular environment,” in: Proceedings of 72nd IEEE Vehicular Technology Conference Fall (VTC-Fall), 2010, pp. 1–5.

[131] S. M. Mousavi, H. R. Rabiee, M. Moshref, and A. Dabirmoghaddam, “Mobisim: A framework for simulation of mobility models in mobile ad-hoc networks,” in: Proceedings of the 3rd International Conference on Wireless and Mobile Computing, Networking and Communications (WiMOB), 2007, pp. 82–82.

[132] J. Härri, F. Filali, C. Bonnet, and M. Fiore, “VanetMobiSim: generating realistic mobility patterns for VANETs,” in: Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, 2006, pp. 96–97.

[133] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, “SUMO Simulation of Urban MObility, An Overview,” in: Proceedings of the 3rd International Conference on Advances in System Simulation, ICST, Brussels, Belgium, 2011, pp. 63–68.

[134] K. C. Lan and C.-M. Chou, “MOVE: A Mobility model generator for Vehicular network,” Retrieved 18-Sep-2014 from: http://www.csie.ncku.edu.tw/~klan/data/materials/mobicom2005.pdf

[135] D. Choffnes and F. Bustamante, “STRAW- An Integrated Mobility and Traffic Model for VANETs,” Retrieved 18-Sep-2014 from: https://pdfs.semanticscholar.org/3bd5/b5407f97af2bbebac3a64481e207e729d185.pdf

[136] J. Miller, “FreeSim–A Free Real-Time V2V and V2I Freeway Traffic Simulator,” Retrieved 18-Sep-2014 from: https://pdfs.semanticscholar.org/b96b/c5b6a404bc78e9afe6c9d5e68f442b03d2cd.pdf

[137] F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, “CityMob: A Mobility Model Pattern Generator for VANETs,” In: Proceedings of the IEEE vehicular networks and application workshop, 2008, pp. 370–374.

[138] F. J. Martinez, C. K. Toh, J.-C. Cano, C. T. Calafate, and P. Manzoni, “A survey and comparative study of simulators for vehicular ad hoc networks (VANETs),” Wireless Communication Mobile Computing, vol. 11, no. 7 (2011), pp. 813–828.

[139] E. Spaho, L. Barolli, G. Mino, F. Xhafa, and V. Kolici, “VANET Simulators: A Survey on Mobility and Routing Protocols,” in: Proceedings of the International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA), 2011, pp. 1–10.

[140] C. Sommer and F. Dressler, “Progressing toward realistic mobility models in VANET simulations,” IEEE Communication Magazine, vol. 46, no. 11 (2008.), pp. 132–137.

[141] F. Bai, N. Sadagopan, and A. Helmy, “The IMPORTANT framework for analyzing the Impact of Mobility on Performance of RouTing protocols for Ad

hoc NeTworks," Elsevier, Ad Hoc Network, vol. 1, no. 4 (Nov. 2003), pp. 383–403.

[142] S. Kurkowski, T. Camp, and M. Colagrosso, "MANET simulation studies: the incredibles," ACM's Mobile Computing and Communication Review, vol. 9, no. 4 (2005), pp. 50–61.

[143] S. Keshav, "REAL: A Network Simulator," University of California at Berkeley, Technical Report No. UCB/CSD 88/472, 1988.

[144] "Monarch Extensions to the ns-2 Network Simulator," Retrieved 4-Jun-2014 from: http://www.monarch.cs.cmu.edu/cmu-ns.html

[145] Q. Chen, D. Jiang, L. Delgrossi, F. S. Eisenlohr, M. T.- Moreno, and H. Hartenstein, "Overhaul of IEEE 802.11 Modeling and Simulation in NS-2." in: Proceedings of 10th ACM/IEEE International Symposium on Modelling Analysis and Simulation of wireless and Mobile System (MSWiM), 2007, pp. 159-168.

[146] K. Hong, J. B. Kenney, V. Rai, and K. P. Laberteaux, "Evaluation of multi-channel schemes for vehicular safety communications," in: Proceedings of 71st IEEE Vehicular Technology Conference (VTC-Spring), 2010, pp. 1–5.

[147] C. Yourong, Tiaojuan ren, Binxing Mo, and Yunwei Lu, "Simulation Environment Realization for Vehicular Ad Hoc Network Based on NS2," Advances in Electric and Electronics, Springer, (2012), pp. 821–828.

[148] A. J. Ghandour, M. Di Felice, L. Bononi, and H. Artail, "Modeling and simulation of WAVE 1609.4-based multi-channel vehicular ad hoc networks," in: Proceedings of 5th International Conference on Simulation Tools and Techniques (ICST), 2012, pp. 148–156.

[149] N. Torabi and B. S. Ghahfarokhi, "Implementation of the IEEE 802.11 p/1609.4 DSRC/WAVE in NS-2," in: Proceedings of 4th International eConference on Computer and Knowledge Engineering (ICCKE), 2014, pp. 519–524.

[150] R. Awdeh, "Evaluating Fairness in Heterogeneous Wireless Ad Hoc Networks," in: Proceedings of 16th International Conference on Computer Communications and Networks (ICCCN), 2007, pp. 1071–1075.

[151] M. D. Laanaoui and P. S. RAGHAY, "The IRUV Protocol: Intelligent Routing Protocol in Urban Environment for Vanet," International Journal of Computer Technology and Applications, vol. 3, no. 2, 2012, pp. 846-850

[152] B. Paul, M. Ibrahim, M. Bikas, and A. Naser, "VANET Routing Protocols: Pros and Cons," International Journal of Computer Applications, vol. 20, no. 3 (2011), pp. 28-34

[153] A. Fonseca and T. Vazão, "Applicability of position-based routing for VANET in highways and urban environment," Journal of Network Computer Application., vol. 36, no. 3 (2013), pp. 961–973.

[154] N. Liu, H. Qian, J. Yan, and Y. Xu, "Performance analysis of routing protocols for vehicle safety communications on the freeway," in: Proceedings of *3rd*

International Conference on Anti-counterfeiting, Security and Identification in Communication, 2009, pp. 85–88.

[155] S. K. ShubhrantJibhkate, A. Kamble, and A. Jeyakumar, “Aodv and Olsr Based Routing Algorithm for Highway and City Scenarios.” International Journal of Advanced Research in Computer and Communication Engineering, vol.4, no. 6 (2015), pp.275-280.

[156] O. Abedi, R. Berangi, and M. A. Azgomi, “Improving Route Stability and Overhead on AODV Routing Protocol and Make it Usable for VANET,” in: Proceedings of 29th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS), 2009, pp. 464–467.

[157] C. Perkins and E. Royer, “Ad hoc On Demand Distance Vector Routing,” in: Proceedings of 2nd IEEE Workshop on. Mobile Computing Systems and Applications, 1999, pp. 90–100.

[158] V. Taliwal, D. Jiang, H. Mangold, C. Chen, and R. Sengupta, “Empirical determination of channel characteristics for DSRC vehicle-to-vehicle communication,” in: Proceedings of 1st ACM international workshop on Vehicular ad hoc networks, 2004, pp. 88–88.

[159] M. Torrent-Moreno, F. Schmidt-Eisenlohr, H. Fussler, and H. Hartenstein, “Effects of a realistic channel model on packet forwarding in vehicular ad hoc networks,” in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), vol. 1, 2006, pp. 385–391.

[160] N. C. Beaulieu and C. Cheng, “Efficient Nakagami-m Fading Channel Simulation,” IEEE Transaction on Vehicular Technology, vol. 54, no. 2 (2005), pp. 413–424.

[161] K. Zhang, Z. Song, and Y. L. Guan, “Simulation of Nakagami Fading Channels with Arbitrary Cross-Correlation and Fading Parameters,” IEEE Transaction on Wireless Communication, vol. 3, no. 5 (2004), pp. 1463–1468.

[162] M. Boban, G. Misek, and O. K. Tonguz, “What is the best achievable QoS for unicast routing in VANETs?,” in: Proceedings of the IEEE GLOBECOM Workshops, 2008, pp. 1–10.

[163] V. Tsaoussidis and I. Matta, “Open issues on TCP for mobile computing,” Journal of Wireless Communication and Mobile Computing, vol. 2, no. 1 (2002), pp. 3–20.

[164] V. Jacobson, R. Braden, and D. Borman, “TCP extensions for high performance.” RFC 1323, Internet Engineering Task Force, 1992.

[165] R. Jain et al., A quantitative measure of fairness and discrimination for resource allocation in shared computer system, Technical Report, ACM Transaction on Computer System, 1984